電腦時代

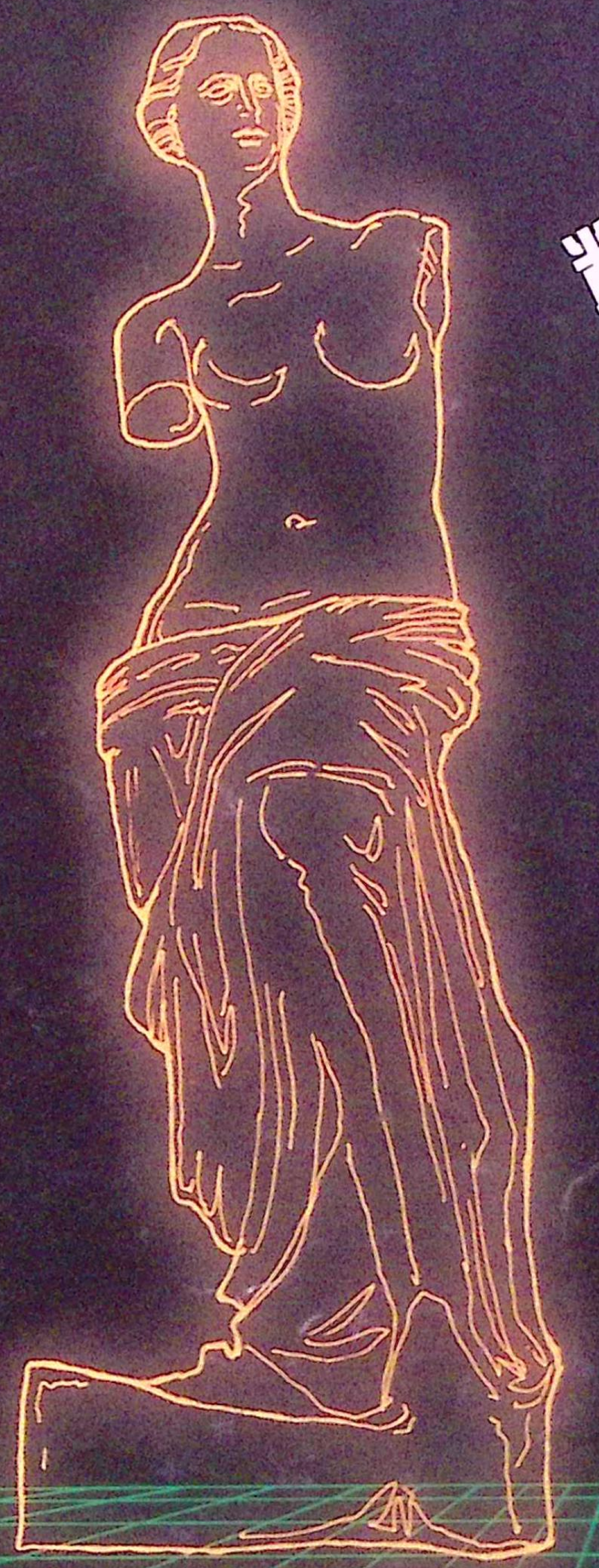
COMPUTING AGE

6

微型電腦用家雜誌

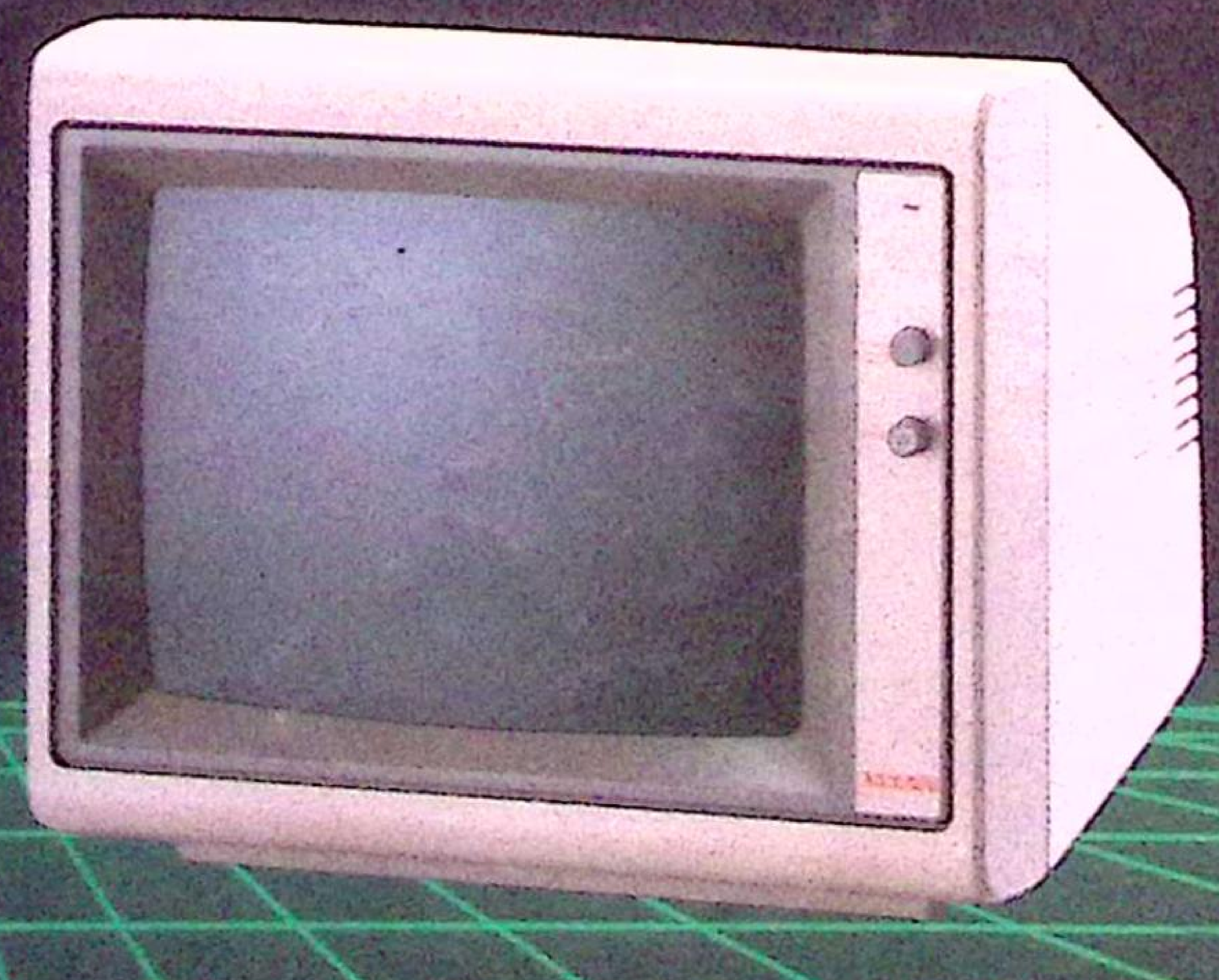
HK$8.00

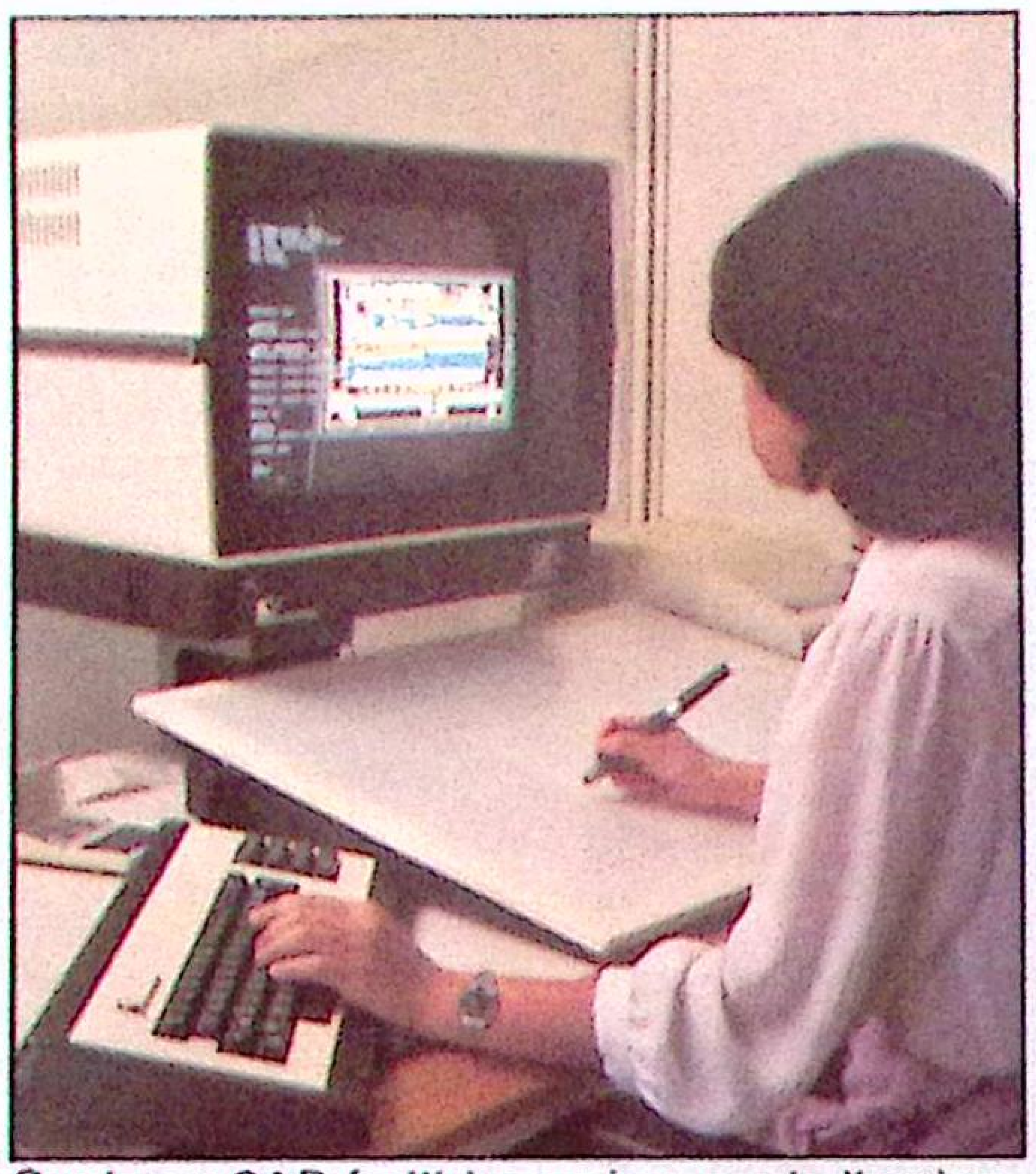
■ 妙用無窮的128 K RAM 咭 ■ 機械人特輯
■ 私人會計系統 ■ APPLESOFT 擴展程式



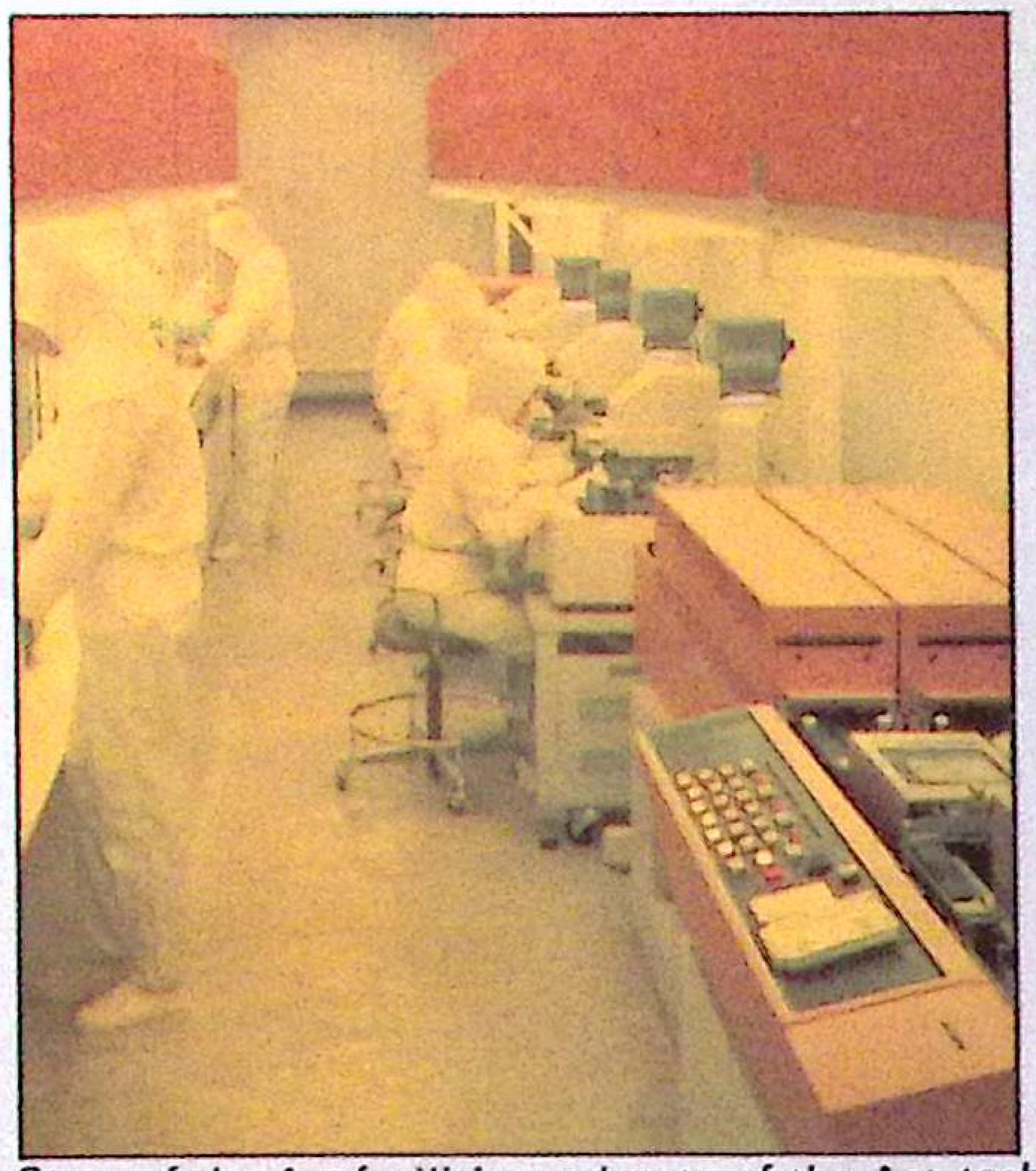
精益求精
擺脫傳統設計
TAXAN® 再放新異彩
全新 KX 系列
綠色電腦螢幕

總代理：
VINCENT COMPUTER CENTRE LTD.
雲生電腦有限公司
九龍尖沙咀柯士甸路22-26號好兆年行608室
電話：3-7211288-0 電訊：HX38316 VCCHK





Our latest CAD facilities are just one indication of our strong commitment to customer design services.





State-of-the-Art facilities and state-of-the-Art technology enable us to provide you with complete silicon-foundry services. Turn-around cycle is just 20 DAYS.

From A New Leader in Silicon Gate CMOS Technology

Our success has been based on manufacturing circuits on the complimentary MOS (CMOS) technology featuring consistently high performance and reliability. Because of the intrinsic superiority in respect to noise immunity, a wide variety of voltage — supply conditions, thrifty use of power, and temperature insensitivity, the CMOS technology is especially ideal for use in harsh environments where no other technology can withstand.
We carry a standard IC line and regularly provide customer design and foundry services. The turn-around cycle for foundry services can be as short as 20 days.

Product Number	Process	Description	TTL or CMOS Compatible	Pin Compatible with
MICROPROCESSOR, MEMORY AND LOGIC				
HKE65SC02	CMOS	8 bit Microprocessor	TTL and CMOS	NMOS 6502
HKE5114	CMOS	1Kx4 Static RAM	TTL and CMOS	2114 4K SRAM
HKE74HCTxxx	CMOS	High Speed CMOS Logic Family	TTL and CMOS	74 Series TTL
HKE74HCxx	CMOS	High Speed CMOS Logic Family	CMOS	74 Series TTL
WATCH CIRCUITS				
HKE151	CMOS	32 KHZ analog	Watch circuit	MB 151
HKE5001	CMOS	3½ digit 5 function multiplexed LCD		MSM 5001

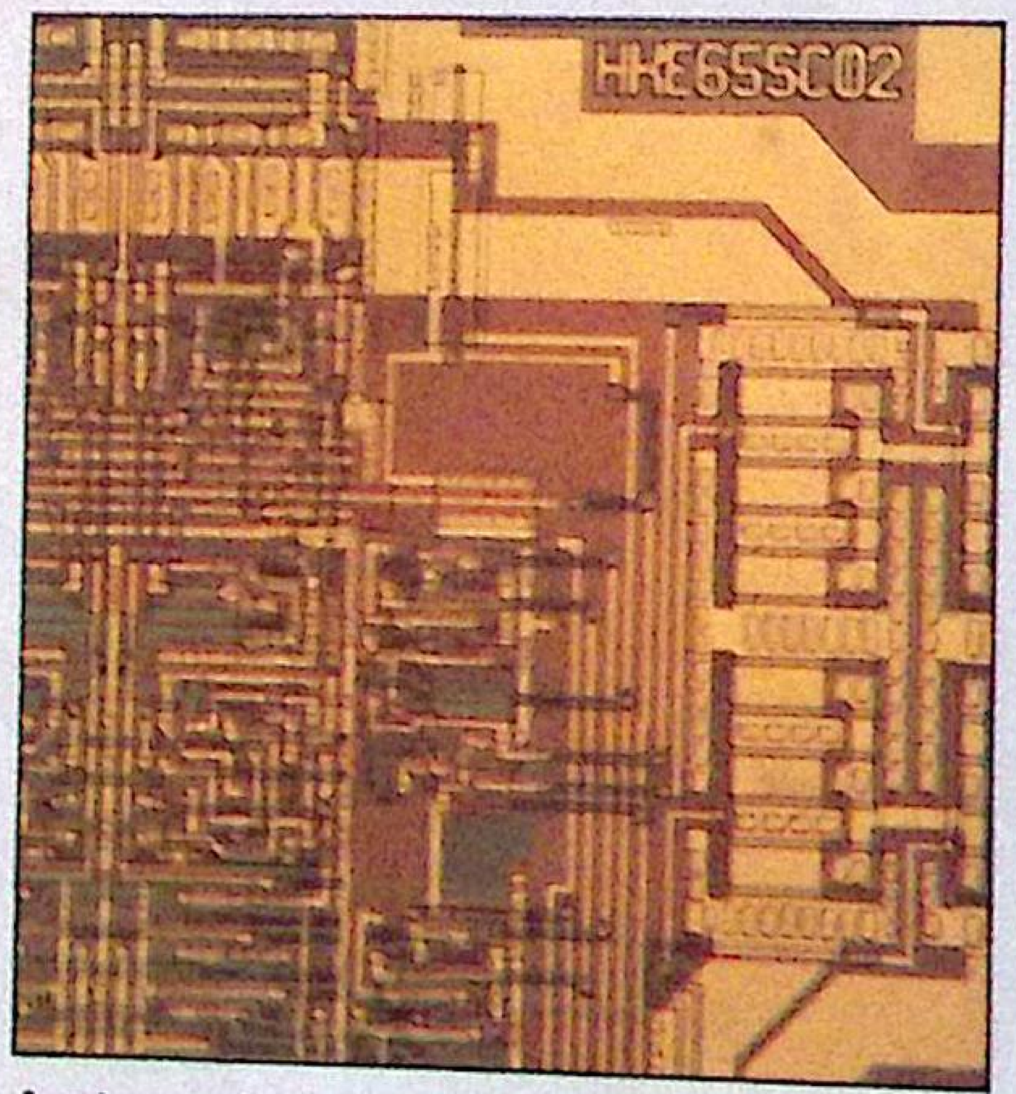
THE SYSTEM DIVISION OF HUA KO
is developing and marketing personal computers and medium size computers for general purpose and turn-key applications.

THE ORGANIC MICRO-modular units for expanding and upgrading Sinclair ZX81 and ZX Spectrum personal computers.

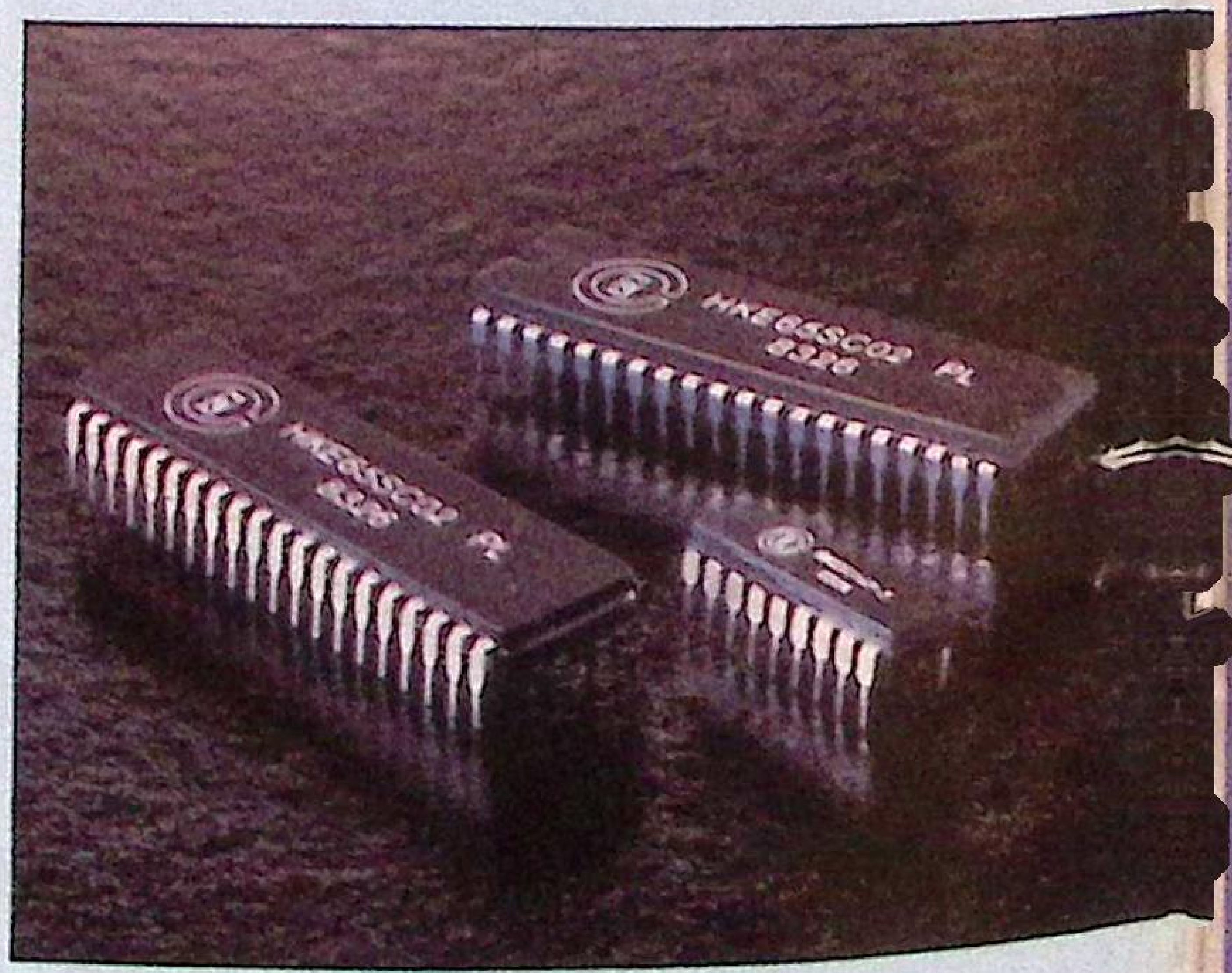
- PERSONA — Interface module enables Organic Mirco to grow on your computer
- MINIMAP — Memory mapping to extend to a maximum of 1M bytes!
- RAM 08 — Low cost memory expandable from 2K to 8K
- RAM 16 — 16K add-on memory at remarkable low cost
- RAM 64 — A TRUE 64K add-on memory
- DROM — For non-volatile storage of programs and data
- TOOLKIT — A module to take 8K bytes of utilities in EPROM/ROM
- PERICON a — Providing 24 lines of input/ output
- PERICON b — 24 lines of heavy duty output-access the outside world
- PERICON c — Drive an 80 column printer with Centronics type interface
- SONUS — Three-voice synthesizer, sound generator with stereo option!
- LINK a — An 8 channels A/D Converter

IBM PC compatible Add-on Memory board
- HKE 64-pc — 16K Bytes
- HKE 128-pc — 128K Bytes
- HKE 192-pc — 192K Bytes
- HKE 256-pc — 256K Bytes

IBM is the registered trademark of International Business Machines.



4 micron design rule and ISO-CMOS silicon gate technology are used in the manufacture of the HKE65SC02.



The very first product announced by HUA KO is an 8 bit CMOS microprocessor, the HKE65SC02, pin-to-pin plug-in compatible with the NMOS 6502.

HUA KO ELECTRONICS CO. LTD.
Head Office & Factory: 9, Dai Shun Street, Tai Po Industrial Estate, Tai Po, N.T., HONG KONG.
TEL: 0-6570373 (10 lines) TELEX: 50898 HKETP HX CABLE: 7599


TAXAN RGB「幻彩系列」
彩色電腦螢幕
色彩艷麗完美無瑕
● 能顯示任何顏色組合
率先採用「線性影像放大系統」
● 絕不干擾磁碟及其他電器
設計符合多項美國嚴格標準
● 適用於各款電腦
備有多種附件，適用於下列電腦：
RGB（紅綠藍）彩色介面
APPLE II¹ APPLE IIe¹（連80字顯示）
介面接駁線
APPLE III¹ IBM PC² BBC³ NEC PC⁴
FUJITSU FM-7⁵ COMMODORE⁶
歡迎訂造特別介面接駁線
1. APPLE IS A TRADE MARK OF APPLE COMPUTER INC.
2. IBM IS A TRADE MARK OF INTERNATIONAL BUSINESS MACHINE CORP.
3. BBC IS A TRADE MARK OF ACORN INTERNATIONAL LTD.
4. NEC PC IS A TRADE MARK OF NEC CORP.
5. FUJITSU FM-7 IS A TRADE MARK OF FUJITSU LTD.
6. COMMODORE IS A TRADE MARK OF COMMODORE ELECTRONICS LTD.
RGB vision-II
TAXAN
享有代理商保證
才是專業性保用
總代理
VINCENT COMPUTER CENTRE LTD.
雲生電腦有限公司
九龍尖沙咀柯士甸路22-26號好兆年行608室 電話：3-7211288 0 電訊：HX38316 VCC HK

LEE'S ELECTRONIC CO. 李氏電子公司

灣仔總店　北角分店

大空型 JOYSTICK

打破傳統設計將兩個發射掣安裝在控制桿上，更易操作，可配合最新推出 JOYSTICK CARD 使用，0-255 讀數可停留在任何位置，AUTO FIRE, 電子控制，可以適應任何新舊 GAME 使用非一般 VR 式可比。

AUTO FIRE掣



ZETA SLIM COMPUTER DATA RECORDER

電腦專用錄音機
獨有 SAVE 及 LOAD 鑑聽和指示
AUTO STOP, 帶長 COUNTER
FOR APPLE COMPUTER
和多種電腦使用，比其他產品更
易 LOAD 帶和 SAVE



"香港區總代理，歡迎批發"

SPEECH CARD II

中文，英文說話卡，更可幫你讀 CATALOG, DOS COMMAND, 外接式高品質小型喇叭箱，聲音響亮清晰 Software + Manual

只售壹佰餘圓

全新推出 JOY-PORT III

只售壹佰餘圓

特別設計全新綫路，絕對可玩全部 GAME
可以選擇 AUTO FIRE 配合 QUICK SHOT JOYSTICK

注意 QUICK SHOT JOYSTICK, 如果唔加 JOYPORT CARD, 唔可以玩 LADY TUT，或者將來新 GAME。

營業時間：上午 11:30 AM 至下午 9:30 PM

香港北角英皇道255號國都商場3樓213室
電話：5-718392

Shop No. 213 Olympia Shopping Centre,
2/F., 255 King's Rd., North Point, H.K.
TEL: 5-718392

香港灣仔道128-150號海運商場1　1-2室
電話：5-8921792

Shop No. 1-2 Ocean Shopping Centre,
1/F., 128-150 Wanchai Rd., Hong Kong.
TEL: 5-8921792

「特惠優待讀者證」

最後申請截止日期六月十五日

「特惠優待讀者證」將於六月中開始寄出，各位讀者不用再來電查詢。而該證的生效日期將於七月開始。歡迎讀者屆時前往讀者服務部參觀選購。未申請的讀者希望你們盡快填妥附載之表格寄來本社。申請截止日期是六月十五日。

自電腦時代出版以來，深受廣大讀者的歡迎，實在令我們獲得鼓舞。而且更重要的是讀者服務部／COMPU HOUSE 能得到許多讀者 的歡迎和支持，我們深深的多謝每一位的讀者。

為著加强滿足和配合讀者的需求，我們決定擴充讀者服務部。搬遷至一個交通更方便的地點和增聘服務人手是我們的第一步。

現時我們正策劃籌備多項的計劃，務求盡力為忠實的擁躉提供更廣泛的多元化服務。其中一項將會是擴大服務部銷貨品種。

為著答謝讀者的愛護，我們特別致送給讀者一張「特惠優待讀者證」。持證者將會獲得我們特別優惠的服務＊（有關詳情，日後將會有所公佈）。

「特惠證」是完全免費送給讀者的，只要填妥下列申請表格，連同寫上姓名地址的回郵信封（貼上四角郵票），寄來九龍中央郵箱71193號便可。

- -

「特惠優待讀者證」申請表格

本人欲申請「特惠優待讀者證」一張。本人無須付出任何費用。

姓名： Lam Lai Ho

年齡： 12

職業： ______

電話： 3-7432347

地址： ______

申請表格可寄往九龍中央郵政信箱71193號電腦時代出版社收
信封面請註明「申請」

此份申請表格請用英文填寫，方便電腦處理。

本人擁有的電腦設備：（請在方格內填✓）

1 6502主機　：一部 ☑　兩部 ☐　三部或以上 ☐

2 5¼吋磁碟機：一部 ☑　兩部 ☐　三部或以上 ☐

3 印字機　：有 ☐　冇 ☑

4 週邊　：80 字行咭 ☐　正考慮購買 ☑
Z－80 咭 ☐　正考慮購買 ☐
ROM咭 ☐　正考慮購買 ☑
說話咭 ☐　正考慮購買 ☑
16K RAM咭 ☐　正考慮購買 ☐
其它咭 1 ______　1 ______
2 ______　2 ______
3 ______　3 ______

5 本人由第 六 期開始閱讀貴刊。

6 本人每期都購買 ☐ ／間中購買 ☑ 電腦時代程式磁碟。

7 程式磁碟是由：特約經銷處購買 ☐
讀者服務部購買 ☐
從朋友處COPY ☑

8 本人是從：報攤 ☑ 發現電腦時代這份雜誌。
電腦公司 ☐
地鐵電視服務站 ☐
代理書局 ☐
高登／黃金商場 ☐
學校圖書館 ☐
公衆圖書館 ☐
朋友口中 ☐

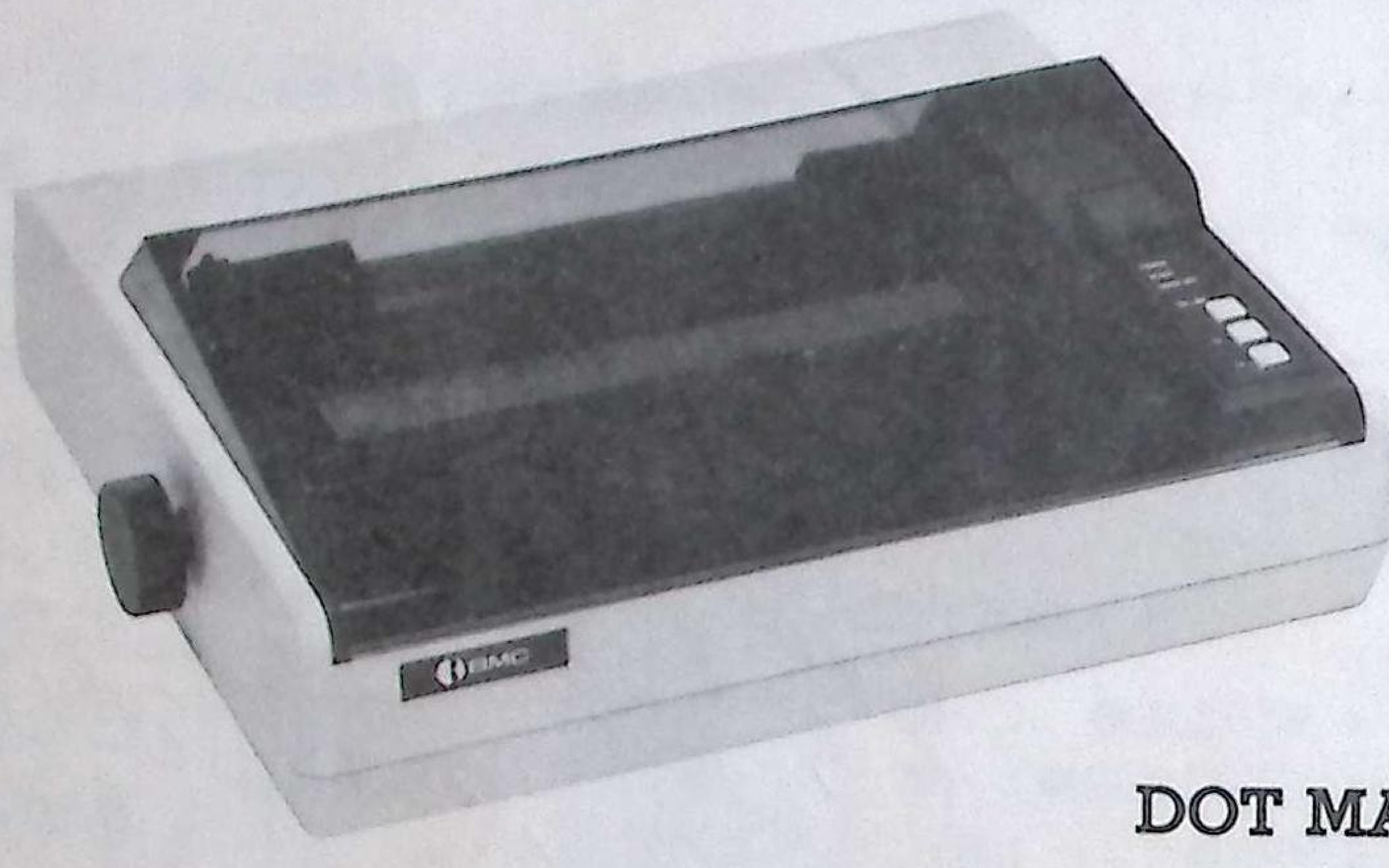
9 假如閣下家中尚未置有磁碟機，是否希望我們製作出版程式錄音磁帶。是 ☐　不感興趣 ☑

10 本人家中使用的磁碟牌子 1 ______
（請填上牌子名稱）2 ______
3 ______

＊註：電腦時代讀者服務部將於短期內擴展服務範圍，並同時增加銷售貨品種類，有關貨品種類，我們將會日後作詳細公佈。我們保證這些貨品售價將會比現時市面最平的售價還要便宜。不過，只有擁有「特惠優待證」的讀者才可享有這些優待。請從速申請 。申請「特惠優待證」是完全免費的。

BMC

超值印字機



BX-80
DOT MATRIX PRINTER

特點：
* 雙向式印字，每秒可印80字。
* 可用普通紙及電腦紙。
* 兩種字體，每行可印71－142大小文字。
* 採用適合之Graphic Interface, 可從Apple* Computer作Graphic Dump.
* 如採用普通紙可印一式四份。
* 標準Centronics平行式接口，適用於一般電腦。

享有代理商保証，才是專業性保用，慎防水貨及二手貨，請索取全新保用証，如証上有任何塗改，即自動作廢。如有疑問，請馬上向本公司查詢。
*APPLE IS A TRADE MARK OF APPLE COMPUTER INC.

*請駕臨各大分銷商參觀



總代理：
雲生電腦有限公司
VINCENT COMPUTER CENTRE LTD.
九龍尖沙咀柯士甸路22·26號好兆年行608室 電話：3-7211288-0 電訊：HX38316 VCC HK

香港電腦研習中心
Hong Kong Centre of Computer Studies

地址： 九龍旺角太子道141號長榮商業大廈506室

電話： 3-804040

本校特點：

一、師資優良：
全部講師爲香港大學畢業或以上程度。持有電腦或有關科目學士或碩士學位，並富有教學經驗。

二、充足實習時間：
置有十餘部電腦，供學員實習。星期一至六，上午十時至晚上八時，學員可隨時回校實習，不另收費。

三、備有教學軟件，學員可利用課餘時間，隨電腦學習。

四、講義由經驗講師精心編寫，深入淺出。

五、交通便利：
本中心位於旺角長榮商業大廈，即大大公司斜對面，地鐵太子站出口，恒生銀行樓上。歡迎參觀及查詢。（星期一至六，上午十時至晚上九時半）。

Introduction to Computer Programming

以教授BASIC電腦語言爲主，並包括電腦知識及磁碟操作系統（DOS）。全期學費連講義及實習共300元。

Advanced BASIC Programming

包括 programming and debugging techniques; CALL, PEEK, POKE; shape table, animated graphics; DOS; CP/M; data file programming; sorting and searching 等。全期學費連講義及實習共320元。

Certificate Course in Computer Business Application

專爲工商界初學電腦人士而設，適合各類型電腦。

1. Introduction to computer application
2. Electronic Spreadsheet
3. Database Management
4. Word Processing

爲期三月，共分兩期，學費每期350元。

Word Processing with WordStar

以電腦代替打字機，處理文字。除WordStar外，兼授Mailmerge及SpellStar，適用於任何能行CP/M之電腦。學費290元。

Database Management with dBASE II

包括用dBASE II處理資料及設計商業用程式，分兩期：

1. Fundamentals of dBASE II, 學費250元。
2. Programming in dBASE II, 學費280元。

本中心另設BASIC函授課程，自編簡明講義，按時寄發，並批改習作，備有錄音帶及教學軟件，學員可隨時回校免費實習。學費180元。

迎接電腦世紀的來臨
學習電腦語言與電腦溝通

RITEMAN 超級印字機

超值售價　超速印字　超薄機身

特點：

- 機身體積爲普通印字機一半。
- 超速印字，每秒可印120字。
- 雙向式穿梭印字，每分鐘可達63行。
- 多種印字功能，可印四國文字，更可作Graphic Dumping（需另加Graphic Interface）。
- 採用新設計之盒式絲質墨帶，用戶可補充油墨重覆使用，操作簡易，墨帶壽命高達一百萬字。
- 減少更換墨帶次數，不斷節省金錢，物超所值。

長期節省開支，半年慳到差價，二十個月慳到一部新機。

	每盒售價	可印字數	每年印字數	需用墨帶	總開支
普　通　墨　帶	HK$60	600,000字	20,000,000字	33盒	HK$1,980
Riteman可補充油墨墨帶	HK$40	1,000,000字	20,000,000字	20盒	HK$ 800

每年節省開支達HK$1,180.

獲美國聯邦通訊委員會，標準檢定合格，証書編號CWT9Z6 RITEMAN，以保証品質。

享有代理商保証，才是專業性保用，愼防水貨及二手貨，請索取全新保用証，如証上有任何塗改，即自動作廢。如有疑問，請立即向本公司

特約經銷商：

Golden Micro Computer Co.
3-793168, 3-602564

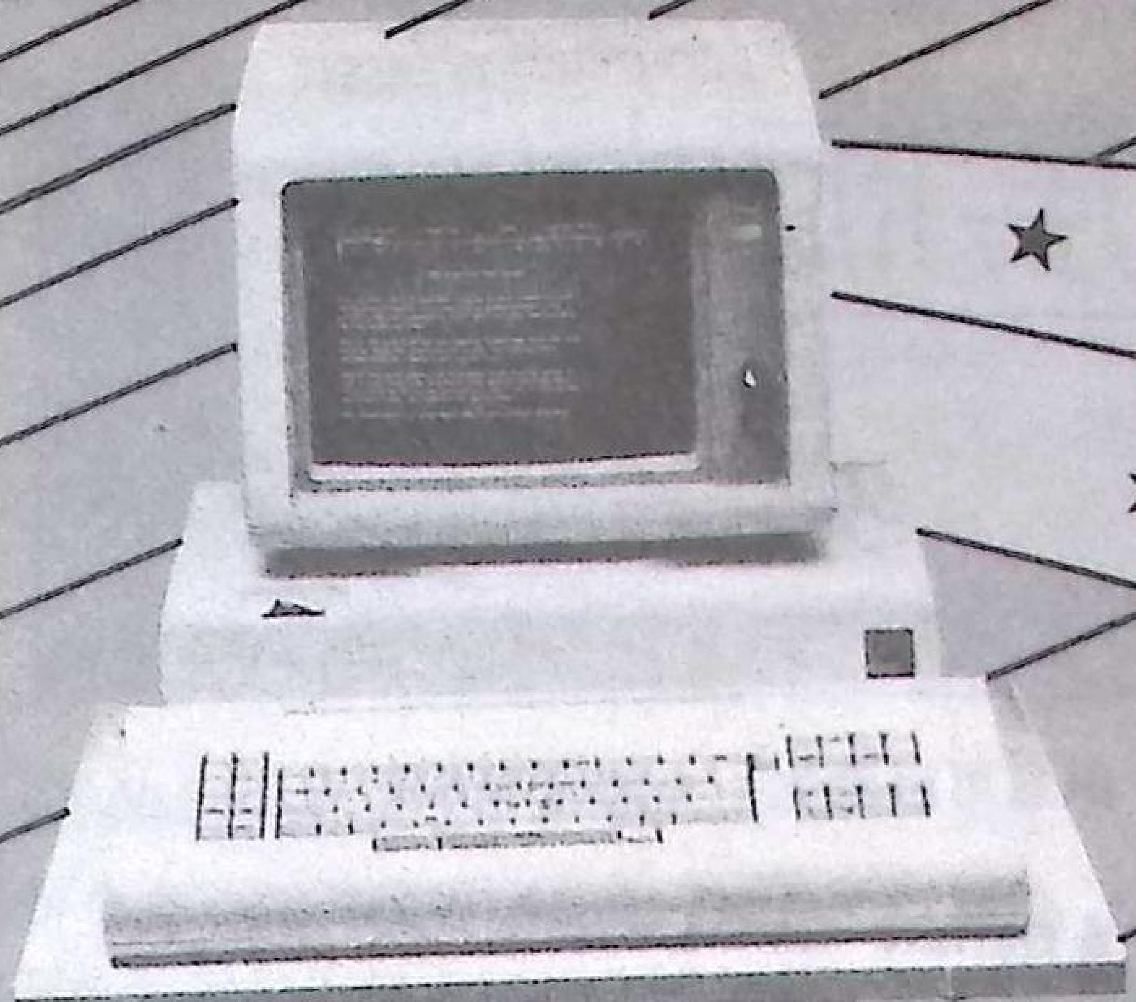
Profit's Computer Co.
3-780983

Sunray Electronics Co.
3-7237390



總代理
VINCENT COMPUTER CENTRE LTD.
(A Subsidiary of Vincent Group)
雲生電腦有限公司
（雲生集團附屬機構）

九龍尖沙咀柯士甸路22-26號好兆年行608室　電話：3-7211288-0
電訊：HX38316 VCC HK



★學習尖端科技

★邁向成功坦途

現代電腦系統訓練中心

MODERN COMPUTER SYSTEM TRAINING CENTRE

——各項電腦課程招生——

COURSE 課程表	上課編制	時數	學費
電腦語言課程			
(1) BASIC PROGRAMMING COURSE	個別或班制	16 HRS	$ 280
(2) ADVANCED BASIC COURSE	個別或班制	15 HRS	$ 300
(3) ASSEMBLER LANGUAGE COURSE	個別或班制	16 HRS	$ 350
(4) C LANGUAGE COURSE	班制上課	16 HRS	$ 800
(5) ADA LANGUAGE COURSE	班制上課	16 HRS	$ 800
電腦硬件課程			
(1) 6502 電腦自嵌實習班	個別或班制	30 HRS	$ 500
(2) Apple II 電腦修理班	班制上課	80 HRS	$1500
(3) IBM PC 電腦修理班	班制上課	80 HRS	$1800
綜合應用課程			
(1) d BASE II 入門及商業應用	個別或班制	16 HRS	$ 400
(2) CP /M 操作系統	個別或班制	16 HRS	$ 350
(3) DOS 應用及拆磁班	班制上課	20 HRS	$ 550
(4) WORDSTAR TRAINING COURSE	個別或班制	16 HRS	$ 350

上述學費概以班制上課爲準，個別教授另議

創辦兩年　電腦碩士教任

本校師資 (1)香港及中文大學電腦系，英美加電腦系畢業生。 (2)英國曼徹斯特大學電腦碩士，英國電腦學會會員。

本中心開放時間：10：00A.M—10：00P.M

地址：九龍深水埗福華街156號黃金商場地庫20號室　　電話：3-601531,3-7289080

分址：荃灣大壩街67號閣樓　　旺角太子道141號長榮商業大厦303A號（彌敦道口）

0-429147　　3-869085

APPLESOFT 擴展版本

□□

APPLESOFT 擴展程式，是將威力巨大的新指令加入 APPLESOFT 。這是新的機械語言程式令 RAM 咭提供指令解譯器及一系列常式，使編寫程式更爲容易。

條件很簡單，你只需有一部蘋果電腦，最少有 64K RAM 。

APPLESOFT 擴展程式的結構定義及代用符號大致上與 APPLESOFT 相同。

以下的說明中， **imm** 表示指令可以用於即時模式（IMMEDIATE MODE），**def** 表示指令可以用於 APPLESOFT 程式。

& < imm & def

將文本窗（TEXT WINDOW）的濶度收縮至 33 字位，使易於容許編輯 APPLESOFT 程式。一如 POKE 33，33：HOME。

& > imm & def

回復文本窗濶度至 40 。一如 POKE 33，40。

& BEGIN name% def

指出程式的開首部分，該程式將會由 & DO指述重覆一次。它必定要尾隨一個變數名稱。如果任何其它變數出現。TYPE MISMATCH ERROR 就會顯示，見 & DO。

& BKGND = aexpr imm & def

將 HI－RES 螢幕轉爲由 AEXPR 選擇的 HCO－LOR 。正常 APPLESOFT 顏色的數字應是由 0 至 7 。8 至 255 數值的螢幕則由不同的形式的顏色塡滿。數字少於 0 及大於 255 會造成 ILLEGAL QUANTITY ERRO－R。

& BELL imm & def

擴音器响一秒，音量是 1 KHZ。

& CALL expr imm & def

容許由十六進制數值召喚匯編語言程式，對 AEXPR ，其功能亦一如正常的召喚（CALL）。在十六進制數值之前必需加上 $ 符號。例如用十六進制標記去CALL 監視 BELL常式，方法有兩種：

```
& CALL $FBDD
```

或

```
A$ = "$FBDD"
& CALL A$
```

& CLRERR imm & def

清除 APPLESOFT ERROR FLAG，使正常錯誤訊息可以產生。一如 POKE ·216，0。

& CLRLN imm & def

將游標位置改到在每一行最尾之處。一如 CALL －868。

& CLRPG imm & def

將游標位置改在螢幕最尾部份，一如 CALL－958。

& DO name% def

即時 JUMP 至緊接於 & BEGIN 有相同 NAME % 的句子之後。它的功能相似於 GOTO ，但執行更爲迅速。因爲它不像 GOTO 需要搜索行數，它的地址已儲存於變數 NAME %。

```
10 X = 1
20 & BEGIN A% : X = X + 1
30 PRINT X
40 IF X < 10 THEN & DO A%
```

在這個例子，當 X ＜10，一個非條件式 JUMP ，會緊接於 & BEGIN A % 句子之後，即X＝X＋1。它不會理會行數的。若 A % 未有下定義，會造成 SYNTAX ERROR。每項 & DO 均有 & BEGIN連結。另方面一個&BEGIN可以與數個 & BEGIN 連結。

& GOTO aexpr imm & def
& GOSUB aexpr imm & def

這容許使用 GOSUB 及 GOTO ，功能一如在 INTEGER BASIC 。句子可用以下形式。

```
& GOTO A * 10 + 20
```

在這個例子，如果A＝10，程式將會 GOTO LINE 120。

& HPRINT aexpr1,aexpr2,expr imm & def
& VPRINT aexpr1,aexpr2,expr imm & def

使用 APPLESOFT TOOL KIT 的字符組格式，此常式會將字符串 EXPR 印在活躍的 HI－RES 螢幕。HI－RES 坐標是 AEXPR 1 ，AEXPR 2。在 NIBBLE.SET（見列表 4）檔案的字符亦可以印出來，甚至你可用 APPLE DOS TOOL KIT 任何的字符組。

左上角第一個字符是根據你所指定的坐標位置印出來，留意印出來的東西是否越出螢幕框範圍。 EXPR 可以打橫（HPRINT）或打直（VPRINT）。字符串或任何算式均可印出，例如，可印出下列。

```
& HPRINT 0,1,"THIS IS A TEST"
& VPRINT 100,99, SIN (B)
& HPRINT 200,40, A$
& HPRINT 0,0, A*42-1
```

最後一行會印出計算結果：A ＊ 42－1

編者按：在讀完本篇文章，了解新的 APPLESOFT 版本後，我們希望讀者嘗試利用這些新指令編寫程式寄來本社。假若程式一經接納及發表刊出，本社將致以薄酬。

投寄的程式在編寫上包含越多的新指令，被選用的機會越大，程式長度不受限制。如程式有一定長度，請將程式 SAVE 入碟（或帶）及連同說明一併寄來九龍中央郵政信箱 71193 號本社收。信封面請註「新指令程式」。

註：※稿件取錄與否，本刊都會退回磁碟或磁帶給讀者。

&INPUT [string;] svar def

這是另個指令，功能相似於 APPLESOFT 指令 IN－PUT 。只用於將單字串輸入指定字串變數，字串會先印出，才會進行 INPUT。當要求 INPUT，可隨意打入任何所需要的字符，包括逗號。整個字串會保留在指定字串變數內。

& MOD aexpr1, aexpr2, avar imm & def

它模仿 INTEGER BASIC 的 MOD 功能，相等於 AEXPR 1 MOD AEXPR 2。結果儲存於 AVAR。AEXPR 1及 AEXPR 2 一定要是整數，如果採用浮點小數或變數，在執行前便要轉作爲整數（INTEGER）。結果會大大不同，出乎你意料之外。例如 & MOD 53，13，X相等於 INTEGER BASIC 敍述句X＝53 MOD 13，計算結果X會是1。

& PEEK expr, var imm & def

這容許 PEEK 在記憶位置上使用十六進制數值。EX－PR 可以是十進制數目，數目變數或是字串。變數會加入記憶位置數值並 PEEK 。如果 VAR 是字串變數，數值是以十六進制標記紀錄，前面並加上＄字符號。例如，如果記憶位置＄4000（16384）含有 $FF（255），可用以下表達方式。：

```
& PEEK 16384,X     X包含 255
& PEEK 16384,X$    X$ 包含 $FF
& PEEK $4000,Z     Z包含 255
```

如果A＝16384 及 B$＝"＄4000"，以下句子亦可接受。

```
& PEEK A,X$        X$ 包含 $FF
& PEEK B$,X        X包含 255
```

& POKE expr,expr[,expr...] imm & def

一如在＆ PEEK，任何十六制及十進制標記均可用於置定記憶位置。如果採用十六進制，之前必須加上"＄"。如果 A$＝"＄300"，B$＝"$FC"，X＝768，Y＝252，以下的表達方式有相同的意思。

```
& POKE 768,252
& POKE $300,252
& POKE A$,$FC
& POKE 768,$FC
& POKE X,B$
& POKE X,Y
& POKE A$,Y
```

注意，任何十六進及十進組合均可容許，額外的表達可加在句子尾部，例如：

```
& POKE $300,$FF,$FE
```

這 $FF 將會被放置在記憶位置 $300，另一個變數放在另一個連續的記憶位置，即是，$FE 在 $301位置。這可排列至 APPLESOFT 容許的最高長度：

```
& POKE $4000,1,2,3,4,5,6,7,8...
```

每個指令只能指定一個記憶地址，而其後的變數一定要在 0－255（十進制）或 $0－$FF （十六進制）之間。

& P imm & def

例子：

```
& P
& PF
& P80
& P132
```

＆P開動印字機（40 欄） APPLE 平行印字機介面置放在 SLOT 1。

＆PF 關上印字機（PR#0）。

＆P 80 開動印字機及校定至指定的欄數。如果欄數是大於 80，印字的濶度將會減少至一吋十六點五個字符（用 PROWRITER或APPLE DOT MATRIX PRIN－TER）。

如果是使用其他牌子印字機，則必須將 53 EB 及 53 F1 稍爲改動。 $53 EB 是壓縮印字（16.5 CPI），$53 F1 是正常印字機模式（10 CPI）。

& RPRINT expr,aexpr [;] imm & def

將指定的內容印出指定次數，假如敍述句尾是分號，則印至尾是不會有回車。

例如：

```
& RPRINT "TEST",40
```

即表示印出"TEST"這個字四十次。

例如：

```
& RPRINT A, 10;
```

印出變數A內容十次。

& WAIT aexpr imm & def

AEXPR指定下，程式執行會暫停數秒。AEXPR 數值可由 0－255。

如果你擁有 APPLE DOS TOOL KIT ，你可用任何 CHARACTER SET 取代 NIBBLE.SET（列表 4）擴展程式的 HPRINT 及 VPRINT 指令將字符印於 HI－RES 螢幕。用家將位置370的 ASC Ⅱ碼檔案名稱改變即可。

例如用 TOOL KIT ASC Ⅱ. SET ，則依循以下步驟去做：

```
BLOAD EXPANDER.OBJ0
CALL -151
370:C1 D3 C3 C9 C9 AE D3 C5 D4 A0
BSAVE EXPANDER.OBJ0,A$300,L$9F
```

APPLESOFT 擴展版本

列表（2）EXPANDER.OBJ 1

```
*8B00L

8B00-   2C 88 C0
8B03-   20 00 D0
8B06-   2C 8A C0
8B09-   60
8B0A-   08
8B0B-   2C 88 C0
8B0E-   28
8B0F-   60
8B10-   8D 8A C0
8B13-   20 98 D9
8B16-   4C 0A 9B
8B19-   8D 8A C0
8B1C-   20 DD FB
8B1F-   4C 0A 9B
8B22-   8D 8A C0
8B25-   20 9C FC
8B28-   4C 0A 9B
8B2B-   8D 8A C0
8B2E-   20 42 FC
8B31-   4C 0A 9B
*L

8B34-   8D 8A C0
8B37-   4C C9 DE
8B3A-   8D 8A C0
8B3D-   20 ED FD
8B40-   4C 0A 9B
8B43-   8D 8A C0
8B46-   20 8E FD
8B49-   4C 0A 9B
8B4C-   8D 8A C0
8B4F-   20 4C E7
8B52-   4C 0A 9B
8B55-   8D 8A C0
8B58-   20 FD E5
8B5B-   4C 0A 9B
8B5E-   8D 8A C0
8B61-   20 E3 DF
8B64-   4C 0A 9B
8B67-   8D 8A C0
8B6A-   20 52 E4
8B6D-   4C 0A 9B
*L

8B70-   8D 8A C0
8B73-   20 E2 E5
8B76-   4C 0A 9B
8B79-   8D 8A C0
8B7C-   20 67 DD
8B7F-   4C 0A 9B
8B82-   8D 8A C0
8B85-   20 52 E7
8B88-   4C 0A 9B
8B8B-   8D 8A C0
8B8E-   20 41 D9
8B91-   4C 0A 9B
8B94-   8D 8A C0
8B97-   20 D6 D3
8B9A-   4C 0A 9B
8B9D-   8D 8A C0
8BA0-   4C D2 D7
8BA3-   8D 8A C0
8BA6-   20 58 FC
8BA9-   4C 0A 9B
*L
8BAC-   8D 8A C0
8BAF-   20 BE DE
8BB2-   4C 0A 9B
8BB5-   8D 8A C0
8BB8-   20 B9 F6
8BBB-   4C 0A 9B
8BBE-   8D 8A C0
8BC1-   20 11 F4
8BC4-   4C 0A 9B
8BC7-   8D 8A C0
8BCA-   20 8A F4
8BCD-   4C 0A 9B
8BD0-   8D 8A C0
8BD3-   20 F8 E6
8BD6-   4C 0A 9B
8BD9-   8D 8A C0
8BDC-   20 5A F4
8BDF-   4C 0A 9B
8BE2-   8D 8A C0
8BE5-   20 4A FF
*L
8BE8-   4C 0A 9B
8BEB-   8D 8A C0
8BEE-   20 3F FF

8BF1-   4C 0A 9B
8BF4-   8D 8A C0
8BF7-   20 CB F5
8BFA-   4C 0A 9B
8BFD-   8D 8A C0
8C00-   20 F4 F3
8C03-   4C 0A 9B
8C06-   8D 8A C0
8C09-   20 F0 F6
8C0C-   4C 0A 9B
8C0F-   8D 8A C0
8C12-   20 A8 FC
8C15-   4C 0A 9B
8C18-   8D 8A C0
8C1B-   20 81 DE
8C1E-   4C 0A 9B
8C21-   8D 8A C0
*L

8C24-   20 C0 DE
8C27-   4C 0A 9B
8C2A-   8D 8A C0
8C2D-   20 3D DB
8C30-   4C 0A 9B
8C33-   8D 8A C0
8C36-   20 5A DB
8C39-   4C 0A 9B
8C3C-   8D 8A C0
8C3F-   20 06 E3
8C42-   4C 0A 9B
8C45-   8D 8A C0
8C48-   20 6F FD
8C4B-   4C 0A 9B
8C4E-   8D 8A C0
8C51-   20 39 D5
8C54-   4C 0A 9B
8C57-   8D 8A C0
8C5A-   20 6C DD
8C5D-   4C 0A 9B
*L

8C60-   8D 8A C0
8C63-   20 12 D4
8C66-   4C 0A 9B
8C69-   8D 8A C0
8C6C-   20 01 E3
8C6F-   4C 0A 9B

8C72-   8D 8A C0
8C75-   20 2B EB
8C78-   4C 0A 9B
8C7B-   8D 8A C0
8C7E-   20 00 E6
8C81-   4C 0A 9B
8C84-   8D 8A C0
8C87-   20 FB E6
8C8A-   4C 0A 9B
8C8D-   8D 8A C0
8C90-   20 7B DD
8C93-   4C 0A 9B
8C96-   8D 8A C0
8C99-   20 34 ED
8C9C-   4C 0A 9B
8C9F-   8D 8A C0
8CA2-   20 3A DB
8CA5-   4C 0A 9B
8CA8-   8D 8A C0
8CAB-   20 7F E9
8CAE-   4C 0A 9B
8CB1-   8D 8A C0
8CB4-   20 66 EA
8CB7-   4C 0A 9B
8CBA-   8D 8A C0
8CBD-   20 A7 E7
8CC0-   4C 0A 9B
8CC3-   8D 8A C0
8CC6-   20 23 EC
8CC9-   4C 0A 9B
8CCC-   8D 8A C0
8CCF-   20 21 EB
8CD2-   4C 0A 9B
*L

8CD5-   8D 8A C0
8CD8-   20 1E EB
8CDB-   4C 0A 9B
8CDE-   8D 8A C0
8CE1-   6C 50 00
8CE4-   60
8CE5-   FF
8CE6-   FF
8CE7-   FF
8CE8-   FF
8CE9-   FF
```

列表（1）EXPANDER.OBJ 0

```
*300L

0300-   A9 00
0302-   8D F6 03
0305-   A9 9B
0307-   8D F7 03
030A-   A9 2C
030C-   CD 00 9B
030F-   D0 10
0311-   A9 88
0313-   CD 01 9B
0316-   D0 09
0318-   A9 C0
031A-   CD 02 9B
031D-   D0 02
031F-   F0 29
0321-   A9 69
0323-   A0 03
0325-   2C 8A C0
0328-   2C 89 C0
032B-   2C 89 C0
032E-   20 3A DB
*L

0331-   A9 83
0333-   A0 03
0335-   20 3A DB
0338-   2C 8A C0
033B-   A9 9A
033D-   8D 01 9D
0340-   20 D4 A7
0343-   A9 4B
0345-   A0 03
0347-   20 3A DB
034A-   60
034B-   8D 84 C2
034E-   CC CF C1
0351-   C4 C5
0353-   D8
0354-   D0 C1
0356-   CE C4 C5
0359-   D2
035A-   AE CF C2
035D-   CA
*L

035E-   B1 AC
0360-   C1 A4
0362-   B9 C2 B0
0365-   B0 A0
0367-   8D 00 8D
036A-   84 C2
036C-   CC CF C1
036F-   C4 CE
0371-   C9 C2
0373-   C2
0374-   CC C5 AE
0377-   D3
0378-   C5 D4
037A-   AC C1 A4
037D-   C4 C4
037F-   B0 B0
0381-   8D 00 8D
0384-   84 C2
0386-   CC CF C1
0389-   C4 C5
*L

038B-   D8
038C-   D0 C1
038E-   CE C4 C5
0391-   D2
0392-   AE CF C2
0395-   CA
0396-   B2
0397-   AC C1 A4
039A-   C4 B0
039C-   B0 B0
039E-   8D 00 FF
03A1-   1B
03A2-   FF
03A3-   1C
```

APPLESOFT 擴展版本

列表 5　所列舉的示範程式是運用擴展程式的初階。使用程式時，先要打入：

```
10 PRINT CHR$(4);"BLOAD EXPANDER.
OBJ0": CALL 768
```

APPLESOFT 擴展程式主要是靠類似 AMPERS－AND 指令解譯器發揮功能。而解譯器則是存放在 RAM咭，指令解譯器分爲三部份：（1）監察指令的密碼；（2）指令表：（3）地址表（見列表 3，位置 $5043 至 50B60）。

不熟習 APPLESOFT 的人，使用 AMPERSAND（＆）字符時，會發覺這字符會製造非條件性的跳躍至 $3F5 位置。駐在該位置的機械語言常式可容許你進行任何的功能。首先，我們需要了解如何解譯指令及如何製造指令以滿足我們的需求。

要認識 AMPERSAND 解譯器內的代碼，關鍵在於 APPLESOFT 採用一個非常重要的零頁指針（ZE－RO POINTER）稱爲 TXTPTR（位署是 $B8 及 $B9）。TXTPTR 的地址用於記憶 APPLESOFT 程式下一個字符，而能爲 BASIC 解譯器所閱讀。

採用 TXTPTR 及 AMPERSAND 矢量，BAS－IS 指令的正常運作會受到干擾，而由 AMPERSA－ND 解譯器取代。

指令表（$5043至$508C）由指令的 ASCⅡ字符之十六進數值構成，每一個指令字串尾部都要加上 $00 而$FF 則用於表的結尾。

例如，BELL 這個字是用 $42，$45，$4C，$4C，$00 代表。如果指令已是 APPLESOFT 保留字，我們可以採用該字的「替代字眼」（TOKEN），GOTO 的「TOKEN」是 $AB，因此在指令表上，GOTO 是用 $AB，$00代表。

深入剖析 各項指令功能

& BELL, & CLRLN and & CLRPG

每一個這些常式均直接用監督程式次程式，在 16 K 上層則無代碼連繫。但由於所有常式均召喚自 16 K 上層，必須在 $5B00 位置經跳躍表，指引它們召喚這三個常式，可依循 $5B10至$5B31 去做。

& CLRERR, & < and & >

& CLRERR在 $5392－$5396，相等於 POKE 216,0.

& ＜ 及& ＞ 在$5296－$52A1 均相當簡單，&＜首先召喚監視程式 HOME 常式，接着壓縮文件窗濶度至 33 字符濶。& ＞ 回復濶度至 40。這兩個指令非常有用。

& GOTO

這常式在 $5269－$5272 。在正常執行的 GOTO 指述，APPLESOFT 使用 LINGET 常式去將程式的行數加入稱爲 LINUM 的零頁指針。但此不能容許運算 GOTO 。使用新常式，JSR FRNUM 取代了LINGET，在 TXTPTR 製造方程式　將數字儲存於浮點累加器（FAC。）JSR GETADER 抽取FAC 內的數字而在 LIN－UM轉化爲兩個 BYTE 的整數。如果召喚 LINGET 之後，馬上輸入 APPLESOFT 常式，GOTO 會正常執行。但現在可以用任何算式去計算行數。GOTO 進入點爲$5941。

& GOSUB

此程式開始一如運算 GOTO 常式。它容許透過使用 FRMNUM 及 CHRGET 運算行數。$5272－$5295 相同於位置$921 的 APPLESOFT 常式。當常式完成工作，就會召喚 NEWSTT 執行下一句子。

& BKGND =

BKGND 常式在 $5441－$544F ，亦是非常簡單。它使用三個 APPLESOFT 常式。首先，跳躍至 GE－TBYT ，它監視由 TXTPTR 所指向的方程式。該方程式產生單 BYTE 的整數及儲存於X － REGISTER 。

接着，HCOLOR 常式會取用X－REGISTER 的數值，及設定$ E4 位置至適當顏色。將這個顏色 BYTE 加入至 A－REGISTER ，就會跳躍至 BKGND，它會使 HI－RES 螢幕出現所選擇的色彩。

HCOLOR 在 0－7 會正常操作。但當大於 7 時，螢幕就會出現有趣圖案。如數值少於零及大於 255，便會顯示 ILLEGAL QUANTITY ERROR。

以下的幾行程式，大家不妨試打入及執行，看看有何結果。

```
10 HGR
20 FOR I=0 TO 255
30 &BKGND = I
40 NEXT
```

&WAIT n

這是簡單常式，令程式暫停數秒。它用於教育程式甚爲有用，可令學習人士清楚了解螢幕所顯示的意思。此常式用 GETBYTE 找尋停頓時間的數值。GETBYTE 儲存數值於 X－REGISTER 。幅度是 0 至 255。

$FCA8的 WAIT 常式，結構甚爲特別，它停頓的時間數值與A－REGISTER 的數值大有差異，在A－RE－GISTER 加上 195（HEX $C3），WAIT 常式會停頓約十分之一秒，若果在Y REGISTER 加入 10（$5453 位置），並將 WAIT 常式重覆十次，可以造成接

列表（3）EXPANDER.OBJ 2

```
*5000L

5000-   A2 00
5002-   86 19
5004-   A0 00
5006-   BD 43 D0
5009-   C9 FF
500B-   F0 20
500D-   D1 B8
500F-   F0 0B
5011-   E8
5012-   BD 43 D0
5015-   D0 FA
5017-   E8
5018-   E6 19
501A-   D0 EA
501C-   E8
501D-   C8
501E-   BD 43 D0
5021-   C9 00
5023-   F0 0B
5025-   D1 B8
*L

5027-   F0 F3
5029-   A0 00
502B-   F0 E4
502D-   4C 34 9B
5030-   20 10 9B
5033-   06 19
5035-   A6 19
5037-   BD 8E D0
503A-   48
503B-   BD 8D D0
503E-   48
503F-   20 B7 00
5042-   60
5043-   42
5044-   45 4C
5046-   4C 00 B5
5049-   00
504A-   84 00
504C-   AB
504D-   00
*L

504E-   B0 00
5050-   43
5051-   4C 52 4C
5054-   4E 00 43
5057-   4C 52 50
505A-   47
505B-   00
505C-   D1 00
505E-   CF
505F-   00
5060-   56 BA
5062-   00
5063-   48
5064-   BA
5065-   00
5066-   43
5067-   4C 52 45
506A-   52
506B-   52
506C-   00
*L

506D-   50 00
506F-   52
5070-   BA
5071-   00
5072-   42
5073-   4B
5074-   47
5075-   4E 44 D0
5078-   00
5079-   E2
507A-   00
507B-   B9 00 4D
507E-   4F
507F-   44
5080-   00
5081-   8C 00 42
5084-   45 47
5086-   49 4E
5088-   00
5089-   44
*L

508A-   4F
508B-   00
508C-   FF
508D-   18
508E-   9B
508F-   4F
5090-   D4
5091-   B6 D0
5093-   68
5094-   D2
5095-   71 D2
5097-   21 9B
5099-   2A
509A-   9B
509B-   94 D2
509D-   9C
509E-   D2
509F-   A1 D2
50A1-   B1 D2
50A3-   91 D3
*L

50A5-   96 D3
50A7-   F8
50A8-   D3
50A9-   43
50AA-   D4
50AB-   FD D0 14
50AE-   D2
50AF-   64
50B0-   D4
50B1-   60
50B2-   D2
50B3-   A3
50B4-   D4
50B5-   C2
50B6-   D4
50B7-   20 3C 9C
50BA-   C9 22
50BC-   D0 0E
50BE-   20 18 9C
50C1-   A9 3B
*L

50C3-   20 21 9C
50C6-   20 2A 9C
50C9-   4C CF D0
50CC-   20 33 9C
50CF-   20 5E 9B
50D2-   20 57 9C
50D5-   A9 3F
50D7-   09 80
50D9-   85 33
50DB-   20 45 9C
50DE-   86 FA
50E0-   20 4E 9C
50E3-   A5 FA
50E5-   20 67 9B
50E8-   A0 02
50EA-   A2 00
50EC-   20 70 9B
50EF-   A5 FA
50F1-   91 83
50F3-   C8
*L

50F4-   A5 6F
50F6-   91 83
50F8-   C8
50F9-   A5 70
50FB-   91 83
50FD-   60
50FE-   20 E3 D1
5101-   C9 2C
5103-   D0 50
5105-   20 B1 00
5108-   A0 00
510A-   B1 50
510C-   A8
510D-   20 69 9C
5110-   20 5E 9B
5113-   24 82
5115-   30 05
5117-   AA
5118-   20 72 9C
511B-   60
*L

511C-   24 81
511E-   30 35
5120-   20 84 9C
5123-   8A
5124-   48
5125-   4A
5126-   4A
5127-   4A
5128-   4A
5129-   20 58 D1
512C-   85 FB
512E-   68
512F-   29 0F
5131-   20 58 D1
5134-   85 FC
5136-   A9 24
5138-   85 FA
513A-   A9 03
513C-   20 67 9B
513F-   A2 FA
*L

5141-   A0 00
5143-   20 70 9B
5146-   A9 03
5148-   91 83
514A-   C8
514B-   A5 6F
514D-   91 83
514F-   C8
5150-   A5 70
5152-   91 83
5154-   60
5155-   4C 34 9B
5158-   09 30
515A-   C9 3A
515C-   90 02
515E-   69 06
5160-   60
5161-   A2 00
5163-   86 3E
5165-   86 3F
*L

5167-   A0 02
5169-   B1 83
516B-   99 FD 00
516E-   88
516F-   10 F8
5171-   C5 06
5173-   F0 07
5175-   30 05
5177-   A2 35
5179-   4C 60 9C
```

近一秒的停頓，過程在 $5450－$545A 進行。

X－REGISTER 儲存停頓多少秒的數值，而我們使用該數值去控制，外迴路（$545F－$5462），故此，你可將停頓程式執行長達255秒。

& BEGIN

此常式可在 $54A4－$54C0位置找出。首先用 E－RRDIR確保並不是處於即時執行模式。如果是的話，就會顯示出 ILLEGAL DIRECT ERROR。

使用 PTRGET索取位於指令之後的變數名稱。$54AA 負責監視變數名稱，確保它們是完整變數。在位置 $54B4－$54BB TXTPTR當時的數值是保留在 VAR－PNT 所指向的區域。

& CALL

& CALL 可在位置 $5261－$5266 找出，但它所用的子程式亦一樣爲& PEEK及& POKE所用。這些子程式在位置 $5194－$5260。& CALL常式的目標找出召喚 LINUM(零頁位置 $50) 內常式的地址。而用非直接的跳躍到地址去，即是 JMP（LINUM）。

& DO

此常式在 $54C3至$54DC位置，功能與& BEGIN 相反。它回復 TXTPTR 較早前爲& BEGIN ，所保留的數值。

& HPRINT X,Y,S
& VPRINT X,Y,S

& HPRINT是藏於位置$52B2至 $5391 一段的記憶裡。兩者能使採用 APPLE TOOL KIT 字符組以X，Y坐標方式寫在高解像螢幕上。HPRINT會將文字橫寫。VPRINT則會將文件垂直寫出。（不過，在本程式中，我們亦一併提供一個字符組，列表4）

APPLE TOOL KIT所提供的 HI－RES 字符產生器，相當有用，但佔去很多記憶。這些常式能主要提供途徑將字符印有 HI－RES 螢幕，第一個字符在螢幕左上角，坐標是X，Y。X，Y是HI－RES 的坐標。

我們可以使用數種 HI－RES 常式，首先，HFNS 索取兩個螢幕坐標，及爲 HPOSN 設置（REGISTER）。即是，REGISTER X存有水平坐標的低次序元數（LOW ORDER BIT）。REGISTER Y存有水平坐標的高次序元數（HIGH－DRDER BIT）。REGIS－TER A 存有垂直螢幕坐標。（X螢幕坐標保存在XSA－V，等待 VPRINT索取，而垂直螢幕坐標則保留在 VS－AVE ，待HPRINT使用）

接着召喚 HPOSN，以較定內外游標，以備 HI－RES常式使用。

隨着X，Y坐標，程式召喚 CHKCOM 以確定是否有豆號（，）存在。在$DD7B位置的 FRMEVL，依照豆號產生的表達意思而將結果儲存在浮點累加器。（FAC）。

檢視$11（STAT）的位置，可以決定 FAC 的結果究竟是字串抑或數字，我們可以使用 FOUT（$52CE）去改變爲字串。接着將字串的地址保留在 STRIA。$52D7－$52E1是用作計算字串的長度。

如果原本的輸出已是字串，召喚在 $52E3 的 FR－ESTR，加入X及Y REGISTER ，及字串的地址。保留在 STR1A與STR1L。

& INPUT

INPUT 常式可找自位置 $50B7－$50FD，開始時召喚 ERRDIR確保並非在即時模式。如果是的話，ILLEGAL DIRECT ERROR 便會出現。

下一步是查驗是否有引號（ASCII $22），如果找出引號，$50BE－50C9 的一段子程式，緊隨其後印出字串。若沒有字串出現，$50CC 就會輸出問號。$50CF召喚 PTRGET索取字串空間 。CHKSTR 確保列出的變數是字串變數。若不是，TYPE MISMATCH ERROR 就會印出。

$50D5－$50D9 設定問號爲 PROMPT 字符。GETLN 索取輸入字串放入輸入緩衝器。

字串的長度保留在 LFOR ，召喚 GDBUF 撤除輸入緩衝器內的高 BIT。在$50E5的 GETSPA 爲新字串在記憶找出空間，MOVSTR 將其移入所需要的記憶位置。剩下來的工作是更改指針至字串，過程是在$50F3至 $50FB 進行。

& MOD

& MOD 則略有不同。語意結構最好做至X＝A MOD B；不過解譯器並不容許如此，故此語意（SYNT－AX）結構只能設計爲& MOD A，B，X，意義一如上述結構。

& MOD 由 $5465開始，用 FRMNUM 以索取第一個數字。INT 轉變它爲整數，MOVIF 移動它至臨時 REGISTER＃1。CHKCOM 檢查豆點。第二個數字亦轉爲整數及移動至臨時 REGISTER＃2。

$547E將第二個數字除第一個數字並召喚 INT 將其轉變爲整數。此整數再用第二個數字相乘，然後，在位置$548F，此結果由第一個數字減除。

使用A及B，可得到一如以下 BASIC 句子的相同結果：

```
INT(A)-(INT(A)/INT(B))*INT(B)
```

APPLESOFT 擴展版本

```
517C-    85 06
517E-    A0 00
5180-    B1 FE
5182-    C9 24
5184-    D0 CF
5186-    C8
5187-    C4 06
5189-    F0 08
518B-    B1 FE
518D-    20 C5 D1
*L

5190-    4C 86 D1
5193-    60
5194-    A2 00
5196-    86 3E
5198-    86 3F
519A-    A0 FF
519C-    C8
519D-    20 B1 00
51A0-    90 0A
51A2-    F0 0E
51A4-    C9 2C
51A6-    F0 0A
51A8-    C9 3B
51AA-    F0 06
51AC-    20 C5 D1
51AF-    4C 9C D1
51B2-    C4 06
51B4-    F0 05
51B6-    30 03
51B8-    4C 77 D1
*L

51BB-    60
51BC-    A5 3F
51BE-    85 51
51C0-    A5 3E
51C2-    85 50
51C4-    60
51C5-    49 30
51C7-    C9 0A
51C9-    90 09
51CB-    69 88
51CD-    C9 FA
51CF-    B0 03
51D1-    4C 34 9B
51D4-    A2 03
51D6-    0A
51D7-    0A
51D8-    0A
51D9-    0A
51DA-    0A
51DB-    26 3E
*L

51DD-    26 3F
51DF-    CA
```

```
51E0-    10 F8
51E2-    60
51E3-    C9 24
51E5-    D0 10
51E7-    A9 04
51E9-    85 06
51EB-    20 94 D1
51EE-    20 BC D1
51F1-    20 B7 00
51F4-    4C 14 D2
51F7-    20 8D 9C
51FA-    24 11
51FC-    10 10
51FE-    A9 05
5200-    85 06
5202-    20 61 D1
5205-    20 BC D1
5208-    20 B7 00
520B-    4C 14 D2
520E-    20 82 9B
5211-    20 B7 00
5214-    60
5215-    20 E3 D1
5218-    C9 2C
521A-    D0 41
521C-    20 B1 00
521F-    C9 24
5221-    D0 0C
5223-    A9 02
5225-    85 06
5227-    20 94 D1
522A-    A5 3E
522C-    4C 44 D2
522F-    20 8D 9C
5232-    24 11
5234-    10 0A
5236-    A9 03
*L

5238-    85 06
523A-    20 61 D1
523D-    4C 2A D2
5240-    20 84 9C
5243-    8A
5244-    A0 00
5246-    91 50
5248-    20 B7 00
524B-    F0 13
524D-    C9 2C
524F-    D0 0C
5251-    E6 50
5253-    D0 02
5255-    E6 51
5257-    20 B1 00
525A-    4C 1F D2
525D-    4C 34 9B
5260-    60
5261-    20 E3 D1
```

```
5264-    D0 F7
*L

5266-    4C DE 9C
5269-    20 79 9B
526C-    20 82 9B
526F-    4C 8B 9B
5272-    20 79 9B
5275-    20 82 9B
5278-    A9 03
527A-    20 94 9B
527D-    A5 B9
527F-    48
5280-    A5 B8
5282-    48
5283-    A5 76
5285-    48
5286-    A5 75
5288-    48
5289-    A9 B0
528B-    48
528C-    20 B7 00
528F-    20 8B 9B
*L

5292-    4C 9D 9B
5295-    20 A3 9B
5298-    A9 21
529A-    85 21
529C-    60
529D-    A9 28
529F-    85 21
52A1-    60
52A2-    20 B5 9B
52A5-    86 FA
52A7-    84 FB
52A9-    48
52AA-    A9 FF
52AC-    85 FE
52AE-    68
52AF-    4C B7 D2
52B2-    20 B5 9B
52B5-    85 FE
52B7-    48
52B8-    A5 E4
*L

52BA-    85 FC
52BC-    A9 7F
52BE-    85 E4
52C0-    68
52C1-    20 BE 9B
52C4-    20 AC 9B
52C7-    20 8D 9C
52CA-    24 11
52CC-    30 15
52CE-    20 96 9C
52D1-    85 19
```

```
52D3-    84 1A
52D5-    A0 FF
52D7-    C8
52D8-    B1 19
52DA-    D0 FB
52DC-    84 1B
52DE-    98
52DF-    F0 33
52E1-    D0 0B
52E3-    20 55 9B
52E6-    86 19
52E8-    84 1A
52EA-    85 1B
52EC-    F0 26
52EE-    A0 00
52F0-    B1 19
52F2-    18
52F3-    E9 1F
52F5-    30 18
52F7-    84 FF
52F9-    A2 DD
52FB-    A0 02
52FD-    18
52FE-    0A
52FF-    90 01
5301-    E8
5302-    88
5303-    10 F8
*L

5305-    18
5306-    86 51
5308-    85 50
530A-    20 19 D3
530D-    A4 FF
530F-    C8
5310-    C4 1B
5312-    D0 DC
5314-    A5 FC
5316-    85 E4
5318-    60
5319-    A0 00
531B-    84 06
531D-    A4 06
531F-    A9 00
5321-    85 FD
5323-    B1 50
5325-    10 04
5327-    C6 FD
5329-    29 7F
*L

532B-    20 67 D3
532E-    E6 06
5330-    A9 08
5332-    C5 06
5334-    F0 0E
5336-    A6 E0
```

指令總表

1、<	8、& CLRLN	15、& INPUT
2、>	9、& CLRPG	16、& MOD
3、& BEGIN	10、& DO	17、& PEEK
4、& BKGND	11、& GOTO	18、& POOK
5、& BELL	12、& GOSUB	19、& P
6、& CALL	13、& HPRINT	20、& RPRINT
7、& CLRERR	14、& VPRINT	21、& WAIT

接着下來，是在 $5492 檢查另一個豆點，經 PTRGET 將指針向着變數，並用 MOVMF 移動它入記憶。

& PEEK

& PEEK 與 MOD 有相同的問題，它的功能是交回一個變數。因此需要成立 & PEEK A,B 格式，A是你想 PEEK 的位置，而B是結果而將會被儲存。& PEEK 與 & CALL 採用相同的子程式。

第一個常式，FIRNUM，在 $50FE 召喚。FIRNUM 索取第一個表達而將其轉變爲 LINUM 內的一個地址。

下一步，是檢查是否有豆號，並將A—REGISTER 連同想 PEEK 的地址內容加入。在 $510D 的 SGNFLT 把Y— REGISTER 的數字轉變爲在 FAC內的一個浮點數字。

在 $5110，使用 PTRGET 索取變數的名稱及地址。如果 UARNAM 指出它並非字串，就用 $5118 的 MOVMF 移動 FAC 至由 PTRGET 找到的變數區域。

如果變數是字串，在 $511C 開始處理。$511C 確保變數並非整個變數，而 $5123—$5138 轉變在 FAC 的數字爲字串，$5143—$5152 移動字串至 VARPNT 所指點的區域，並更改所需要的指針。注意任何十六進或十進制數值均被容許。

& POKE

& POKE 與 PEEK 差不多，同樣在 $5215 召喚 FIRNUM 加入四個位數字到 LINUM。$521F—$5243 的代碼，工作形式一如 FIRNUM ，所不同者，下一個變數一定是十進數的 0—255 或 16 進制 $00—$FF。計算的結果放在A — REGISTER，而 $5246 儲存 POKE 數值在 LINUM 所指的記憶位置。

在 $524D，檢查最後的字符是否有豆點，如找到，LINUM 的地址處增1並迴路返回 MORNUM 索取另一個變數，POKE 入下一個地址。

這令加入 APPLESOFT 的簡短機械語言程式非常輕易，無須轉變它爲十進制數。例如，在 $529D—52A1 的回恢螢幕常式可用以下的 BASIC 句子加在開始位置 $300。

```
& POKE $300,$A9,$28,$85,$21,$60
```

& P

& P是用作保留幾個鍵掣，（KEYSTROKE），當用動在 SLOT 1 的印字機及接連上蘋果平衡介面。在 $5397，檢查（F）。如果找到"F"，$539B—$53A9 重新較定CSW 開關掣至 COUT 監督常式及 $FDF0 位置。召 $53AA至 $53B2 較定CSW 開關掣至印字機地址 $100，如果你的印字機地址與此不同，則要更改該段位置的數值，以反映該地址。

同樣地 DOS HOOK 亦在 $53B2 更改。如果指令只是& P，可終止常式。如果在P以後，有欄數，可在 $53BA—$53BC 向介面咭發出 CONTROL—1。$53C1—$53D5 跟隨P以後，輸送每一個字符直接到介面咭。在 $53D8一個"N"輸送入平衡咭。

最後，在 $53E5— $53EA，每一個 ESC 及 $4E 送往印字機以較定每吋印十個字符。如果欄數大於80，$53F0 較定印字密度爲每吋十六點五個字符（16.5CPI）。& 80 是打出 PR#1—CTRL 1—80 N 的捷徑。由於程式是連接入 DOS HOOK ，所以用此指令時，無需加入 CONTROL—D。

& RPRINT

此常式可找自 $53F9—$5441。當你輸入程式，TXTPTR指着跟隨 PRINT 之後的字符。這可以是變數或表達方式。FRMEVL 產生方程式，由 TXTPTR 指點，並將結果放置浮點累加器（FAC）。

BIT STAT 檢查結果是數字抑或是串字，如果 STAT是正數，可交由NUM 處理此數字。反之，召喚 FREFAC，把字串地址放在X及Y REGISTER，及將字串長度放在A—REGISTER。

APPLESOFT EXPANDER 並沒有佔用全部上層 16K 的 BANK 1 空間，所以我們仍有足夠位置加上任何上述的功能。由 $D4DE—$DCFF 仍是空置的。

$8B00 位置的跳躍表，差不多已用盡兩頁的空間。DOS 緩衝器更可移遠些，只要更改 $33 B就可以獲得更多的頁數。

APPLESOFT 擴展程式共包涵有五個列表

列表（1）EXPANDER.OBJ 0
列表（2）EXPANDER.OBJ 1
列表（3）EXPANDER.OBJ 2
列表（4）字符組（NIBBLE.SET）
列表（5）示範程式（EXPANDER DEMO）

■

```
5338-    A4 E1
533A-    E6 E2
533C-    A5 E2
533E-    20 BE 9B
5341-    4C 1D D3
5344-    A9 FF
5346-    C5 FE
5348-    D0 0E
534A-    A6 FA
534C-    A4 FB
534E-    E6 E2
5350-    A5 E2
5352-    20 BE 9B
5355-    4C 63 D3
*L

5358-    A6 E0
535A-    A4 E1
535C-    A5 FE
535E-    20 BE 9B
5361-    E6 E5
5363-    20 F4 9B
5366-    60
5367-    20 E2 9B
536A-    A5 45
536C-    29 01
536E-    F0 0F
5370-    A4 E5
5372-    20 D9 9B
5375-    24 FD
5377-    10 06
5379-    B1 26
537B-    09 80
537D-    91 26
537F-    A5 45
5381-    4A
*L

5382-    F0 0A
5384-    85 45
5386-    A4 E5
5388-    20 C7 9B
538B-    4C 6A D3
538E-    20 EB 9B
5391-    60
5392-    A9 00
5394-    85 D8
5396-    60
5397-    C9 46
5399-    D0 0F
539B-    A9 FD
539D-    85 37
539F-    A9 F0
```

```
53A1-    85 36
53A3-    20 EA 03
53A6-    20 B1 00
53A9-    60
53AA-    A9 00
*L

53AC-    85 36
53AE-    A9 C1
53B0-    85 37
53B2-    20 EA 03
53B5-    20 B7 00
53B8-    F0 3B
53BA-    A9 89
53BC-    20 3A 9B
53BF-    A0 00
53C1-    B1 B8
53C3-    F0 13
53C5-    C9 3A
53C7-    F0 0F
53C9-    B0 2B
53CB-    C9 30
53CD-    90 27
53CF-    09 80
53D1-    20 3A 9B
53D4-    C8
53D5-    4C C1 D3
*L

53D8-    A9 4E
53DA-    09 80
53DC-    20 3A 9B
53DF-    20 43 9B
53E2-    20 D0 9B
53E5-    A9 1B
53E7-    20 3A 9B
53EA-    A9 4E
53EC-    E0 51
53EE-    30 02
53F0-    A9 51
53F2-    20 3A 9B
53F5-    60
53F6-    4C 34 9B
53F9-    20 8D 9C
53FC-    24 11
53FE-    10 3A
5400-    20 7B 9C
5403-    86 FC
5405-    84 FD
*L

5407-    A8
5408-    A9 00
```

```
540A-    F0 02
540C-    B1 FC
540E-    99 00 02
5411-    88
5412-    10 F8
5414-    A9 02
5416-    85 FC
5418-    A9 00
541A-    85 FD
541C-    20 4C 9B
541F-    86 06
5421-    A4 FC
5423-    A5 FD
5425-    20 9F 9C
5428-    C6 06
542A-    D0 F5
542C-    20 B7 00
542F-    C9 3B
*L

5431-    D0 03
5433-    4C B1 00
5436-    4C 43 9B
5439-    60
543A-    20 96 9C
543D-    84 FC
543F-    85 FD
5441-    4C 1C D4
5444-    20 D0 9B
5447-    20 06 9C
544A-    A5 E4
544C-    20 FD 9B
544F-    60
5450-    20 D0 9B
5453-    A0 0A
5455-    A9 C3
5457-    20 0F 9C
545A-    88
545B-    C0 00
545D-    D0 F6
*L

545F-    CA
5460-    E0 00
5462-    D0 EF
5464-    60
5465-    20 79 9B
5468-    20 C3 9C
546B-    20 CC 9C
546E-    20 AC 9B
5471-    20 79 9B
5474-    20 C3 9C
5477-    20 D5 9C
```

```
547A-    A9 93
547C-    A0 00
547E-    20 B1 9C
5481-    20 C3 9C
5484-    A9 98
5486-    A0 00
5488-    20 A8 9C
548B-    A9 93
548D-    A0 00
*L

548F-    20 BA 9C
5492-    20 AC 9B
5495-    20 5E 9B
5498-    24 82
549A-    10 03
549C-    4C BE D4
549F-    AA
54A0-    20 72 9C
54A3-    60
54A4-    20 3C 9C
54A7-    20 5E 9B
54AA-    A5 82
54AC-    10 10
54AE-    A5 81
54B0-    10 0C
54B2-    A0 01
54B4-    A5 B8
54B6-    91 83
54B8-    88
54B9-    A5 B9
*L

54BB-    91 83
54BD-    60
54BE-    A2 A3
54C0-    4C 60 9C
54C3-    20 3C 9C
54C6-    20 5E 9B
54C9-    A5 82
54CB-    10 F1
54CD-    A5 81
54CF-    10 ED
54D1-    A0 01
54D3-    B1 83
54D5-    85 B8
54D7-    88
54D8-    B1 83
54DA-    85 B9
54DC-    60
54DD-    00
54DE-    00
54DF-    00
```

列表（5）示範程式（EXPANDER DEMO）

```
10  REM  **********************
20  REM  * LANGUAGE  EXPANDER *
30  REM  * BY COMPUTING AGE   *
40  REM  * COPYRIGHT (C) 1984 *
70  REM  **********************
80  PRINT  CHR$ (4);"BLOADEXPANDE
     R.OBJO": CALL 768: DIM T(50)
90  HOME : PRINT "** COPYRIGHT 19
     84 BY COMPUTING AGE    **": PRINT
     "THIS IS A DEMONSTRATION OF
     SOME OF THE"
100  PRINT "FEATURES AVAILABLE ON
      THE RAM CARD"
110  PRINT "LANGUAGE EXPANDER."
120  &  WAIT 4
130  PRINT : PRINT : PRINT "FIRST
      WE USE &BELL TO RING THE BE
     LL."
140  &  WAIT 3
150  FOR I = 1 TO 10: & BELL: NEXT
160  VTAB 10: PRINT : PRINT : PRINT
     "NOW USE &CLRLN TO CLEAR TO
     THE EOL AND  DELETE AN EXPLE
     TIVE    %#&*!!@#$%!@!!"
170  &  WAIT 4
180  VTAB 13: HTAB 22: & CLRLN: &
      WAIT 3
190  VTAB 14
200  PRINT : PRINT "NOW TO LINE 4
      AND CLEAR PAGE (&CLRPG)."
210  &  WAIT 3
220  VTAB 4: & CLRPG
230  &  WAIT 3: HOME
240  PRINT "&< WILL SHRINK THE TE
     XT WINDOW TO 33"
250  PRINT "CHARACTERS WIDE TO AL
     LOW EASY EDITING."
260  &  WAIT 4
270  &  < : LIST 10,100
280  &  WAIT 4: &  > : PRINT : PRINT
     .
290  PRINT "&> RESTORES WINDOW TO
      NORMAL.": PRINT
300  &  WAIT 3: LIST 10,100
310  &  WAIT 3: HOME
320  PRINT "YOU CAN INPUT ANYTHIN
     G WITH &INPUT."
330  PRINT : PRINT "COMMAS, COLON
     S, OR ANY OTHER 'ILLEGAL'  C
     HARACTERS ARE OKAY."
340  &  WAIT 2: PRINT
350  PRINT : &  INPUT "TRY IT! IN
     PUT ANYTHING:";X$: PRINT : PRINT
     "SEE, NO 'EXTRA IGNORED'!!!"
360  &  WAIT 2: PRINT : PRINT "HE
     RE IS WHAT YOU KEYED IN: "
370  PRINT : PRINT X$
380  &  WAIT 3: HOME
390  HOME : PRINT "WE CAN ALSO PR
     INT A STRING MANY TIMES."
400  PRINT : PRINT "INPUT A SHORT
      STRING...";: &  INPUT X$
410  PRINT : PRINT "HOW MANY TIME
     S TO PRINT IT..";: INPUT A
420  PRINT : & R PRINT X$,A
430  PRINT : PRINT "THIS WAS DONE
      WITH &RPRINT."
440  &  WAIT 5
450  &  WAIT 5
460  HOME : PRINT "DO YOU HAVE A
     PRINTER  ";: INPUT A$
470  IF  LEFT$ (A$,1) <  > "Y" THEN
     570
480  PRINT : PRINT "DO YOU HAVE A
      PARALLEL CARD IN SLOT 1 ";:
      INPUT A$
490  IF  LEFT$ (A$,1) <  > "Y" THEN
     570
500  PRINT : PRINT "GOOD...TURN T
     HE PRINTER ON PLEASE."
510  &  WAIT 5: PRINT : PRINT "&P
     WILL TURN THE PRINTER ON."
520  &  WAIT 2: & P
530  PRINT : PRINT "&PF WILL TURN
      IT OFF."
540  &  WAIT 2: & PF
550  PRINT : PRINT "&P80 AND &P13
     2 WILL TURN THE PRINTER ON W
     ITH 80 OR 132 COLUMNS"
560  &  WAIT 5
570  HGR : HOME : VTAB 23
580  PRINT "&BKGND= WILL FILL THE
      HI-RES SCREEN WITH";
590  PRINT "THE HCOLOR SPECIFIED.
     "
600  FOR I = 0 TO 7: & BKGND = I:
      VTAB 23: HTAB 24: & CLRLN: PRINT
     "&BKGND = ";I: &  WAIT 1: NEXT
610  PRINT : PRINT "NUMBERS GREAT
     ER THAN SEVEN GIVE"
620  PRINT "INTERESTING RESULTS."
630  &  WAIT 2: FOR I = 100 TO 10
     5: & BKGND = I: VTAB 23: HTAB
     24: & CLRLN: PRINT "&BKGND =
     ";I: &  WAIT 2: NEXT
640  PRINT : PRINT : PRINT "WE CA
     N ALSO WRITE CHARACTERS ON"
650  PRINT "THE HI-RES SCREEN AT
     ANY LOCATION."
```

```
660  &  WAIT 5: PRINT : PRINT "TH
     E UPPER LEFT HAND CORNER OF
     THE STRING";
670  PRINT "WILL BE AT THE X,Y CO
     ORDINATES SELECTED.";: &  WAIT
     5
680  HOME : VTAB 22: PRINT "WHAT
     X COORDINATE DO YOU WANT:";:
      INPUT X
690  PRINT "NOW THE Y COORDINATE:
     ";: INPUT Y
700  PRINT "NOW WHAT DO YOU TO SA
     Y..";: &  INPUT A$
710  & BKGND = 0: & H PRINT X,Y,A
     $
720  &  WAIT 4: & BKGND = 0
730 A$ = "INCREMENTING X BY 1 AND
      DOUBLE PRINTING"
740 B$ = "BRINGS INTERESTING RESU
     LTS."
750  & H PRINT 0,50,A$: & H PRINT
     0,60,B$
760  &  WAIT 4: & H PRINT 1,50,A$
     : & H PRINT 1,60,B$
770  &  WAIT 5: & BKGND = 0: PRINT
     : PRINT : PRINT
780  PRINT "WE MAY ALSO PRINT VER
     TICALLY.": &  WAIT 3
790  & BKGND = 0: PRINT "HERE IS
     THE VALUE OF SIN(3)"
800  & V PRINT 160,0, SIN (3): &
      WAIT 5
810  TEXT
820  HOME
830  PRINT "COMPUTED GOTO'S AND G
     OSUB'S ARE OK"
840  PRINT : PRINT "&GOTO 10*X IS
      A LEGAL STATEMENT"
850 X = 87: &  GOTO 10 * X: &  WAIT
     4
860  PRINT "OOP'S COMPUTED GOTO D
     OSEN'T WORK"
870  PRINT : PRINT "WE ARRIVED HE
     RE VIA COMPUTED GOTO!": &  WAIT
     5
880  HOME : PRINT : PRINT "&POKE
     IS CHANGED TO ALLOW HEX INPU
     T"
890  PRINT "THE FORMAT IS:"
900  PRINT : PRINT "    &POKE $30
     0,$FF"
910  PRINT : PRINT "ANY COMBINATI
     ON OF DECIMAL AND HEX "
920  PRINT "IS ALLOWABLE....."
930  PRINT : PRINT "THE ONLY REST
     RICTION IS THAT HEX MUST BE"
     ;
940  PRINT "PRECEDED BY A '$' (IT
      MAY BE STORED AS  A STRING.
     "
950  PRINT : PRINT "FOR EXAMPLE,
     &POKE A$,255 IS LEGAL"
960  PRINT "WHERE A$= $0300"
970  PRINT : PRINT
980  PRINT : &  INPUT "TYPE A TWO
      DIGIT HEX NUMBER:";A$
990  IF  LEFT$ (A$,1) = "$" THEN
     1010
1000  PRINT : PRINT "REMEMBER TO
     PRECEDE THE NUMBER BY A $": GOTO
     980
1010  PRINT : PRINT "THIS NUMBER
     IS NOW PUT IN MEMORY BY:"
1020  PRINT : PRINT "        POK
     E $0300,A$"
1030  &  POKE $0300,A$
1040  PRINT : PRINT "WHERE A$ IS
     THE NUMBER YOU TYPED."
1050  &  WAIT 5
1060  HOME : PRINT : PRINT "WE CA
     N RECOVER THE NUMBER WITH &P
     EEK."
1070  PRINT : PRINT "WITH &PEEK $
     300, X$ WE CAN READ"
1080  PRINT "MEMORY LOCATION $030
     0 AND STORE IT"
1090  PRINT "IN VARIABLE LOCATION
      X$."
1100  PRINT : PRINT "BY PRINTING
     X$, WE CAN SEE THE HEX VALUE
     ";
1110  PRINT "OF LOCATION $0300."
1120  PRINT : PRINT "HIT THE SPAC
     E BAR TO SEE.": GET S$
1130  PRINT : &  PEEK $0300,X$: PRINT
     "&PEEK $0300, X$=";X$
1140  &  WAIT 5: PRINT : PRINT "A
     NY COMBINATION OF HEX OR DEC
      WILL WORK"
1150  &  WAIT 4
1160  PRINT : PRINT "& CALL WORKS
      THE SAME AS PEEK AND POKE."
     : PRINT "WE MAY CALL A ROUTI
     NE USING HEX        NOTATIO
     N."
1170  PRINT : PRINT "FOR EXAMPLE,
      THE BELL ROUTINE AT $FBDD "
1180  PRINT "MAY BE CALLED WITH &
     CALL $FBDD.": &  WAIT 6
1190  PRINT : PRINT "LIKE THIS ":
      &  CALL $FBDD
1200  &  WAIT 3: HOME : PRINT " T
     HERE IS ALSO A MOD FUNCTION.
```

```
1210  PRINT : PRINT "THE FORMAT I
     S & MOD X,Y,A ": PRINT
1220  PRINT "THIS IS THE SAME AS
     THE INTEGER BASIC ": PRINT "
     STATEMENT....": PRINT
1230  HTAB 10: PRINT "A = X MOD Y
     "
1240  PRINT : INPUT "TRY IT NOW..
     .INPUT AN INTEGER:";X
1250  PRINT : INPUT "NOW INPUT AN
     OTHER INTEGER LESS THAN THE
     FIRST:";Y
1260  & MODX,Y,A
1270  PRINT : PRINT X;" MOD ";Y;"
      = ";A
1280  &  WAIT 4
1290  HOME : PRINT " THE & BEGIN
     AND & DO CAN SAVE SOME "
1300  PRINT "EXECUTION TIME ": &
      WAIT 3
1310  PRINT : PRINT " WE WILL NOW
      EXECUTE A LOOP USING GOTO "

1320  PRINT : PRINT "WHEN YOU ARE
      READY HIT ANY KEY"
1330  GET S$:X = 1
1340 X = X + 1:Y =  SIN (X): HOME
     : PRINT X
1350  IF X < 300 THEN 1340
1360  & BELL: PRINT : PRINT "LOOP
      DONE"
1370  &  WAIT 4: PRINT : PRINT "N
     OW WE WILL EXECUTE THE SAME
     LOOP "
1380  PRINT "USING &BEGIN AND & D
     O"
1390  PRINT "HIT ANY KEY WHEN YOU
      ARE READY": GET S$
1400 X = 1: & BEGINA%
1410 X = X + 1:Y =  SIN (X): HOME
     : PRINT X
1420  IF X < 300 THEN  & DOA%
1430  & BELL: PRINT : PRINT "LOOP
     DONE"
1440  PRINT : PRINT "THERE... THA
     T SHOULD HAVE BEEN FASTER"
1450  PRINT "BY ALMOST 2 SECONDS!
     !"
1460  &  WAIT 5
1470  HOME : PRINT "WELL, THAT'S
     ABOUT IT..."
1480  &  WAIT 3
1490  PRINT : PRINT "EXCEPT FOR W
     AIT......"
1500  PRINT : PRINT "ALL OF THE D
     ELAYS WERE GENERATED BY"
1510  PRINT : PRINT "& WAIT T  WH
     ERE T IS NUMBER OF SECONDS."

1520  &  WAIT 4:
1530  & BELL: PRINT : PRINT "THAN
     KS FOR WATCHING!!"
```

列表（4）字符組（NIBBLE.SET）

```
*9000.9300

9000- 00 00 00 00 00 00 00 00
9008- 08 08 08 08 08 00 08 00
9010- 14 14 14 00 00 00 00 00
9018- 14 14 3E 14 3E 14 14 00
9020- 08 3C 0A 1C 28 1E 08 00
9028- 06 26 10 08 04 32 30 00
9030- 04 0A 0A 04 2A 12 2C 00
9038- 08 08 08 00 00 00 00 00
9040- 08 04 02 02 02 04 08 00
9048- 08 10 20 20 20 10 08 00
9050- 08 2A 1C 08 1C 2A 08 00
9058- 00 08 08 3E 08 08 00 00
9060- 00 00 00 00 00 08 08 04
9068- 00 00 00 3E 00 00 00 00
9070- 00 00 00 00 00 00 08 00
9078- 00 20 10 08 04 02 00 00
9080- 1C 22 32 2A 26 22 1C 00
9088- 08 0C 08 08 08 08 1C 00
9090- 1C 22 20 18 04 02 3E 00
9098- 3E 20 10 18 20 22 1C 00
90A0- 10 18 14 12 3E 10 10 00
90A8- 3E 02 1E 20 20 22 1C 00
90B0- 18 04 02 1E 22 22 1C 00
90B8- 3E 20 10 08 04 04 04 00
90C0- 1C 22 22 1C 22 22 1C 00
90C8- 1C 22 22 3C 20 10 0C 00
90D0- 00 00 08 00 08 00 00 00
90D8- 00 00 08 00 08 08 04 00
90E0- 10 08 04 02 04 08 10 00
90E8- 00 00 3E 00 3E 00 00 00
90F0- 04 08 10 20 10 08 04 00
90F8- 1C 22 10 08 08 00 08 00
9100- 1C 22 2A 3A 1A 02 3C 00
9108- 08 14 22 22 3E 22 22 00
9110- 1E 22 22 1E 22 22 1E 00
9118- 1C 22 02 02 02 22 1C 00
9120- 1E 22 22 22 22 22 1E 00
9128- 3E 02 02 1E 02 02 3E 00
9130- 3E 02 02 1E 02 02 02 00
9138- 3C 02 02 02 32 22 3C 00
9140- 22 22 22 3E 22 22 22 00
9148- 1C 08 08 08 08 08 1C 00
9150- 20 20 20 20 20 22 1C 00
9158- 22 12 0A 06 0A 12 22 00
9160- 02 02 02 02 02 02 3E 00
9168- 22 36 2A 2A 22 22 22 00
9170- 22 22 26 2A 32 22 22 00
9178- 1C 22 22 22 22 22 1C 00
9180- 1E 22 22 1E 02 02 02 00
9188- 1C 22 22 22 2A 12 2C 00
9190- 1E 22 22 1E 0A 12 22 00
9198- 1C 22 02 1C 20 22 1C 00
91A0- 3E 08 08 08 08 08 08 00
91A8- 22 22 22 22 22 22 1C 00
91B0- 22 22 22 22 22 14 08 00
91B8- 22 22 22 2A 2A 36 22 00
91C0- 22 22 14 08 14 22 22 00
91C8- 22 22 22 14 08 08 08 00
91D0- 3E 20 10 08 04 02 3E 00
91D8- 3E 06 06 06 06 06 3E 00
91E0- 00 02 04 08 10 20 00 00
91E8- 3E 30 30 30 30 30 3E 00
91F0- 00 00 08 14 22 00 00 00
91F8- 00 00 00 00 00 00 00 7F
9200- 04 08 10 00 00 00 00 00
9208- 00 00 1C 20 3C 22 3C 00
9210- 02 02 1E 22 22 22 1E 00
9218- 00 00 3C 02 02 02 3C 00
9220- 20 20 3C 22 22 22 3C 00
9228- 00 00 1C 22 3E 02 3C 00
9230- 18 24 04 1E 04 04 04 00
9238- 00 00 1C 22 22 3C 20 1C
9240- 02 02 1E 22 22 22 22 00
9248- 08 00 0C 08 08 08 1C 00
9250- 10 00 18 10 10 10 12 0C
9258- 02 02 22 12 0E 12 22 00
9260- 0C 08 08 08 08 08 1C 00
9268- 00 00 36 2A 2A 2A 22 00
9270- 00 00 1E 22 22 22 22 00
9278- 00 00 1C 22 22 22 1C 00
9280- 00 00 1E 22 22 1E 02 02
9288- 00 00 3C 22 22 3C 20 20
9290- 00 00 3A 06 02 02 02 00
9298- 00 00 3C 02 1C 20 1E 00
92A0- 04 04 1E 04 04 24 18 00
92A8- 00 00 22 22 22 32 2C 00
92B0- 00 00 22 22 22 14 08 00
92B8- 00 00 22 22 2A 2A 36 00
92C0- 00 00 22 14 08 14 22 00
92C8- 00 00 22 22 14 08 08 06
92D0- 00 00 3E 10 08 04 3E 00
92D8- 38 0C 0C 06 0C 0C 38 00
92E0- 08 08 08 08 08 08 08 08
92E8- 0E 18 18 30 18 18 0E 00
92F0- 2C 1A 00 00 00 00 00 00
92F8- 7F 7F 7F 7F 7F 7F 7F 7F
9300- 00
```

私人會計系統

程傑英

雖然坊間有許多功能強大的財政會計程式，但對於私人一般用途來說未免有點大材小用。而且這等程式在使用上都要花一段時間熟習。

筆者在這裡介紹的一個會計程式則非常適合個人採用作收支紀錄，程式能就閣下私人的信用咭、支票及現金花費作詳細紀錄。程式能將這些輸入，透過適當的指令指示，分別整理出三個不同的報告、格式及十個不同的家庭開支報告。

程式使用時相當方便，不會像坊間的會計程式那般複雜，令人摸不著頭腦。

當 RUN 程式後，螢幕首先會出現如下：

```
        HOME FINANCE MANAGEMENT
=====================================

  THIS IS YOUR PERSONAL SYSTEM FOR
MANAGING CREDIT CARDS, CHECKS AND
OVERALL  HOME FINANCES.

YOU'LL BE ASKED TO SELECT FROM THESE
OPTIONS:
          ADD
          POST
          SORT
          REPORT
          END THE SESSION

TRAC WILL MAINTAIN YOUR EXPENSE DATA
IN THE FOLLOEING FILES:

          CREDIT CARD FILE
          CHECK FILE
          YEAR-TO-DATE FILE
          POSTED CREDIT CARD FILE
          POSTED CHECK FILE
HIT RETURN TO CONTINUE
```

按 RETURN，螢幕會顯示：

```
            * MAIN MENU *
ENTER THE NUMBER OF THE OPTION YOU WANT
TO USE.
          1 = ADD RECORDS
          2 = POST (DELETE) RECORDS
          3 = SORT RECORDS
          4 = REPORT/LIST OPTIONS
          5 = END THE SESSION
ENTER YOUR SELECTION: 1
```

在這情況，由於未曾有任何紀錄入案，故可選擇第1項：ADD.RECORDS：螢幕會顯示：

```
ENTER THE NUMBER OF THE FILE TO BE
USED.

          1 = CREDIT CARDS
          2 = CHECKS
ENTER YOUR SELECTION :1
```

我們要求製造一個信用咭 FILE，程式系統會要求首先輸入月份、日子、年份的資料。（在輸入時不可忘記在各數字之間加插一個逗號。按 RETURN，螢幕會顯示出完整的會計賬項及指示用法：

```
          1. AUTO EXPENSE
          2. BOOKS/MAGS.
          3. BUSINESS EXP
          4. CLOTHES ADULT
          5. CLOTHES KIDS
          6. DONATIONS
          7. ELECTRICITY
          8. ENTERTAINMENT
          9. FOOD
          10. GIFTS
          11. HOBBY-ADULT
          12. HOBBY-KIDS
          13. HOME-GAS
          14. HOME-MAINT'
          15. INSURANCE
          16. INVESTMENTS
          17. LIQ/CIG'S
          18. MORTGAGE
          19.TELEPHONE
          20.TAX-INCOME
          21.TAX-COMPANY
          22.TAX-INVEST.
          23.VACATION
          24.WATER

1=MC  2=VSA  3=AMEX  4=GAS
5=OTHER
TYPE ACCOUNT\, 'REDO', 'EDIT' OR
'END'
```

信用咭的輸入包括如下：
1＝MASTERCHARGE
2＝VISA
3＝AMERICAN EXPRESS
4＝GAS
5＝OTHER

（假如上述的信用咭並未包括閣下所擁有在內，可改動行句43及550，輸入閣下擁有的信用咭名稱。）

按著便可輸入有關賬目。假如在輸入後發覺有錯誤，可按 REDO 以便重新再輸入。當輸入完後，打入 END，程式便會自動產生一個 FILE 存入磁碟。其他支票及現金開支輸入方法亦一樣。（若有現金轉賬，閣下大可將之歸入信用咭 FILE 的 OTHER CATALOG ，或使用一個固定的支票號碼（如999，作爲現金轉賬）。

在 SORTING時，螢幕會有下列的顯示：

```
        * SORT CARD # RECORDS *

WHICH FIELD DO YOU WANT TO SORT ON?
        1 = ACCOUNT NUMBER
        2 = CR CARD/CHECK NUMBER
        3 = MONTH
        4 = DOLLAR AMOUNT

ENTER YOUR SELECTION: 3
```

當輸入整個過程完筆後，程式會自動將所有的變數及螢幕清除，等候下一件工作。

在編印報告時，螢幕的指示會如下：

```
     **REPORT OPTION**
1=LIST AND TOTAL CREDIT CARD FILE
2=LIST AND TOTAL CHECK FILE
3=TREND ANALYSIS&AVERAGE MONTH
4=YEAR TO DATE SPENDING PROFILE
5=HI-RES SPENDING GRAPE
6=LIST POSTED CREDIT CARD FILE
7=LIST POSTED CHECK FILE
```

最後，讓我們來討論如何將檔案中的數據 DELETE。當閣下選擇這項目時，螢幕的顯示如下：

```
DELETING A RECORD WILL AUTOMATICALLY
ADD IT TO THE BALANCE BEING CARRIED
FORWARD FOR TREND REPORTING

WHICH RECORD# DO YOU WANT TO DELETE ?
ENTER RECORD # OR TYPE 'END':
```

必須緊記，閣下所 DELETE 的紀錄是以 RECO—#及 SEQUENCE #代表。當輸入指定的 REORD #，系統會自動將該紀錄的名稱（NAME）及總數（AM—OUNT）顯示出來。

〔對於每月整理更新信用咭檔案，將信用咭歸類入 CARD # 及 SEQUENCE ，然後編印出及存入磁碟。

當 DELETE 紀錄時，有以下的事件出現：被刪除的紀錄會在 LIST 內消失；LIST 內各項目會重新再排序。DOLLAR AMOUNT 會自動重新計算出每一個新數值。

本會計系統可幫助閣下了解每月的開支情況，包括：

（1）私家車
（2）雜誌
（3）商業開支
（4）成人服裝開支
（5）小童服裝開支
（6）捐贈
（7）電費
（8）娛樂
（9）食物
（10）禮物
（11）興趣消費（成人）
（12）興趣消費（小孩）
（13）煤氣
（14）家居維修
（15）保險
（16）投資
（17）烟酒
（18）供屋
（19）電話
（20）稅項
（21）水電

系統除能詳細紀錄各項開支，還可以將這些開支作出一個綜合收支趨勢報告，清楚顯示閣下的收支情況是否出現問題！

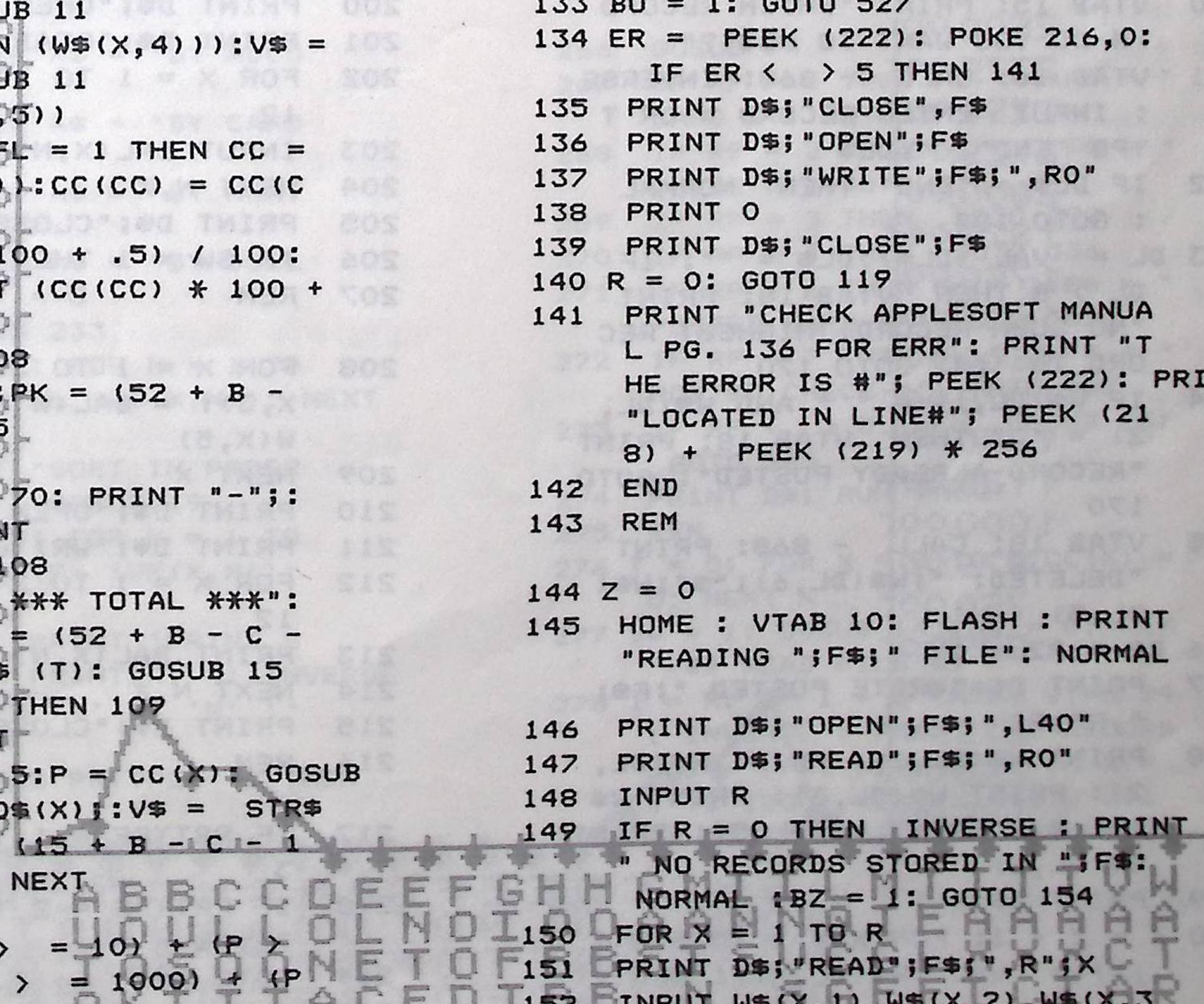
希望閣下能好好利用本會計報告系統。

本程式在程式磁碟的 FILENAME 是 PERSONAL ACCOUNT SYSTEM（簡稱PAS）和PASG。■

```
10  GOTO 486
11  REM
12  PRINT  SPC( PK - SY)V$;
13 SY = SY + (PK - SY) +  LEN (V$
   )
14  RETURN
15  REM
16  PRINT  SPC( PK - SY)V$
17 PK = 0:SY = 0
18  RETURN
19 M = R
20 M =  INT (M / 2): IF M = 0 THEN
   35
21 J = 1:K = R - M
22 H = J
23 V = H + M
24  IF WK(H,SR) < WK(V,SR) THEN 3
   2
25  FOR F = 1 TO 5
26 TK = WK(H,F):WK(H,F) = WK(V,F)
   :WK(V,F) = TK
27  NEXT
28 T$ = W$(H,6):W$(H,6) = W$(V,6)
   :W$(V,6) = T$
29 H = H - M
30  IF H < 1 THEN 32
31  GOTO 23
32 J = J + 1
33  IF J > K THEN 20
34  GOTO 22
35  FLASH : PRINT "SORT COMPLETED
   ": NORMAL
36  RETURN
37  REM
38  HOME : HTAB 8: INVERSE : PRINT
   "* ADD "F$;" RECORDS *": NORMAL
39  FOR N = 1 TO 12: PRINT N$(N);
40  HTAB 20: PRINT N$(N + 12)
41  NEXT N
42  PRINT
43  IF FL = 1 THEN  INVERSE : PRINT
   "1=MC  2=VSA  3=AMEX  4=GAS
   5=OTHER"
44  IF FL = 2 THEN  INVERSE : PRINT
   "ENTER CHECK# 999 FOR CASH E
   NTRY"
45  PRINT "TYPE ACCOUNT*, 'REDO',
   'EDIT' OR 'END'": NORMAL
46  PRINT
47  POKE 34, PEEK (37): RETURN
48  RETURN
49  REM
50  GOSUB 37
51 I = I + 1: NORMAL
52  VTAB 19: CALL  - 868: PRINT "
   REC #";I
53  VTAB 20: CALL  - 868: INVERSE
   : INPUT "ACCOUNT # ";W$(I,1)
54  GOSUB 74
55  IF W$(I,1) = "END" THEN  NORMAL
   :I = I - 1: RETURN
56  IF W$(I,1) = "REDO" THEN I =
   I - 1: GOTO 52
57  IF W$(I,1) = "EDIT" THEN IS =
   I: INPUT "RECORD # TO EDIT?"
   ;I:W$(IS,1) = "": GOTO 52
58 W =  VAL (W$(I,1)): IF W > 24 OR
   W < 1 THEN  GOSUB 75: GOTO 5
   3
59  VTAB 21: CALL  - 868: PRINT F
   $;: INPUT "";W$(I,2)
60  GOSUB 74
61 W =  VAL (W$(I,2)): IF FL = 1 AND
   (W < 1 OR W > 5) THEN  GOSUB
   75: GOTO 59
62  VTAB 21: HTAB 25: CALL  - 868
   : INPUT "MONTH,DAY:";W$(I,3)
   ,W$(I,4)
63  GOSUB 74
64 WM =  VAL (W$(I,3)):WD =  VAL
   (W$(I,4)): IF WM > 12 OR WD >
   31 THEN  GOSUB 75: GOTO 62
65  VTAB 22: CALL  - 868: PRINT "
   PAID TO: ";: NORMAL : PRINT
   "--------------";: HTAB 10: INPUT
   "";W$(I,6): INVERSE
66  GOSUB 74: IF W$(I,6) = "" THEN
   W$(I,6) = " "
67  IF  LEN (W$(I,6)) > 14 THEN  VTAB
   23: PRINT "MAX 14 CHARS ALLO
   WED..PLEASE REENTER": GOTO 6
   5
68  VTAB 22: HTAB 25: CALL  - 868
   : INPUT "AMOUNT:";W$(I,5)
69  GOSUB 74: IF  LEN (W$(I,5)) =
   0 THEN  PRINT "": GOTO 68
70  IF  ASC (W$(I,5)) < 48 OR  ASC
   (W$(I,5)) > 57 THEN 20: PRINT
   "NOT A NUMBER..PLEASE REENTE
   R": GOTO 68
71  HOME : PRINT "LAST ACCT#";W$(
   I,1);" CD#"W$(I,2);" DT ";W$
   (I,3);"/";W$(I,4);" AMT $";W
   $(I,5)
72  IF IS > 0 THEN X = I: GOSUB 1
   21:I = IS:IS = 0: GOTO 52
73  GOTO 51
74  VTAB 23: CALL  - 868: RETURN
75  VTAB 23: PRINT "INVALID ENTRY
   ..PLEASE REENTER": RETURN
76  REM
77 YY = T:T = 0: IF XX THEN T = Y
   Y
78  PRINT D$;"PR#1"
```

```
79  PRINT  CHR$ (9);"100N"
80  PRINT  CHR$ (2); CHR$ (30)
81  PRINT  TAB( 25)F$;" SUMMARY R
    EPORT"
82  PRINT  TAB( 27)"TODAY'S DATE
    ";MT;"/";D;"/";Y
83  IF OM = 3 THEN  PRINT  TAB( 2
    7)"SORTED ";A$
84 FF$ =  RIGHT$ (F$,7)
85  PRINT :PK = 1:V$ = "SEQ": GOSUB
    11:PK = 12:V$ = "** PAID TO
    **": GOSUB 11:PK = 28:V$ = "
    ACC": GOSUB 11:PK = 36:V$ =
    FF$: GOSUB 11:PK = 44:V$ = "
    MO": GOSUB 11:PK = 50:V$ = "
    DAY": GOSUB 11
86 PK = 56:V$ = "AMOUNT": GOSUB 1
    5
87  FOR X = 1 TO 70: PRINT "=";: NEXT
    X: PRINT
88  FOR X = 1 TO I
89 PK = 1:V$ =  STR$ (X): GOSUB 1
    1
90 PK = 9:V$ = W$(X,6): GOSUB 11
91 PK = (31 -  LEN (W$(X,1))):V$ =
    W$(X,1): GOSUB 11
92 PK = (41 -  LEN (W$(X,2))):V$ =
    W$(X,2): GOSUB 11
93 PK = (46 -  LEN (W$(X,3))):V$ =
    W$(X,3): GOSUB 11
94 PK = (52 -  LEN (W$(X,4))):V$ =
    W$(X,4): GOSUB 11
95 W =  VAL (W$(X,5))
96 T = T + W: IF FL = 1 THEN CC =
    VAL (W$(X,2)):CC(CC) = CC(C
    C) + W
97 T =  INT (T * 100 + .5) / 100:
    CC(CC) =  INT (CC(CC) * 100 +
    .5) / 100
98 P = W: GOSUB 108
99 V$ =  STR$ (W):PK = (52 + B -
    C1): GOSUB 15
100  NEXT X
101  FOR X = 1 TO 70: PRINT "-";:
     NEXT X: PRINT
102 P = T: GOSUB 108
103 PK = 1:V$ = " *** TOTAL ***":
     GOSUB 11:PK = (52 + B - C -
     1):V$ =  STR$ (T): GOSUB 15
104  IF FL < > 1 THEN 109
105  PRINT : PRINT
106  FOR X = 1 TO 5:P = CC(X): GOSUB
     108: PRINT CD$(X);:V$ =  STR$
     (CC(X)):PK = (15 + B - C - 1
     ): GOSUB 15: NEXT
107  GOTO 109
108 B = 9:C = (P >  = 10) + (P >
     = 100) + (P >  = 1000) + (P
     >  = 10000): RETURN
109  PRINT  CHR$ (9);"40N"
110  PRINT D$;"PR#0"
111  IF SW = 1 THEN  RETURN
112  TEXT : CLEAR :BU = 1: GOTO 4
     87
113  REM
114  ONERR  GOTO 134
115  PRINT D$;"OPEN";F$;",L40"
116  PRINT D$;"READ";F$;",RO"
117  INPUT R
118  PRINT D$;"CLOSE";F$
119 I = R: GOSUB 49
120  TEXT
121  PRINT D$;"OPEN";F$;",L40"
122  IF IS > 0 THEN 126
123  PRINT D$;"WRITE";F$;",RO"
124  PRINT I
125  FOR X = R + 1 TO I
126  PRINT D$;"WRITE";F$;",R";X
127  PRINT W$(X,1): PRINT W$(X,2)
     : PRINT W$(X,3): PRINT W$(X,
     4): PRINT W$(X,5): PRINT W$(
     X,6)
128  IF IS > 0 THEN 130
129  NEXT X
130  PRINT D$;"CLOSE";F$
131  IF IS > 0 THEN  RETURN
132  IF SW = 1 THEN  RETURN
133 BU = 1: GOTO 527
134 ER =  PEEK (222): POKE 216,0:
     IF ER <  > 5 THEN 141
135  PRINT D$;"CLOSE",F$
136  PRINT D$;"OPEN";F$
137  PRINT D$;"WRITE";F$;",RO"
138  PRINT 0
139  PRINT D$;"CLOSE";F$
140 R = 0: GOTO 119
141  PRINT "CHECK APPLESOFT MANUA
     L PG. 136 FOR ERR": PRINT "T
     HE ERROR IS #"; PEEK (222): PRINT
     "LOCATED IN LINE#"; PEEK (21
     8) +  PEEK (219) * 256
142  END
143  REM
144 Z = 0
145  HOME : VTAB 10: FLASH : PRINT
     "READING ";F$;" FILE": NORMAL
146  PRINT D$;"OPEN";F$;",L40"
147  PRINT D$;"READ";F$;",RO"
148  INPUT R
149  IF R = 0 THEN  INVERSE : PRINT
     " NO RECORDS STORED IN ";F$:
     NORMAL :BZ = 1: GOTO 154
150  FOR X = 1 TO R
151  PRINT D$;"READ";F$;",R";X
152  INPUT W$(X,1),W$(X,2),W$(X,3
```

```
2. BOOKS/MAGAZINES          $  442.00
3. BUSINESS EXP.            $  246.00
4. CLOTHES (ADULT)          $  244.00
5. CLOTHES (CHILDREN)       $  123.00
6. DONATIONS                $    0.00
7. ELECTRICITY              $    0.00
8. ENTERTAINMENT            $    0.00
9. FOOD                     $    0.00
10. GIFTS                   $    0.00
11. HOBBIES (ADULT)         $    0.00
12. HOBBIES (CHILDREN)      $    0.00
13. HOME-GAS                $    0.00
14. HOME-MAINT.             $    0.00
15. INSURANCE               $    0.00
16. INVESTMENTS             $    0.00
17. LIQUOR                  $    0.00
18. MORTGAGE                $    0.00
19. TELEPHONE               $    0.00
20. TAX-INCOME              $    0.00
21. TAX-COMPANY             $    0.00
22. TAX-INVEST.             $    0.00
23. VACATION                $    0.00
```



```
    ),W$(X,4),W$(X,5),W$(X,6)
153  NEXT X
154  PRINT D$;"CLOSE";F$
155  IF SW = 1 THEN  RETURN
156  ONERR  GOTO 160
157  PRINT D$;"OPEN POSTED ";F$;"
     ,L40"
158  PRINT D$;"READ POSTED ";F$;"
     ,RO"
159  INPUT R2: GOTO 165
160 ER =  PEEK (222): POKE 216,0:
      IF ER <  > 5 THEN 141
161  PRINT D$;"CLOSE POSTED ";F$
162  PRINT D$;"OPEN POSTED ";F$;"
     ,L40"
163  PRINT D$;"WRITE POSTED ";F$;
     ",RO"
164  PRINT 0:R2 = 0
165  PRINT D$
166  HOME : HTAB 8: PRINT "* POST
     ";F$;" RECORDS *"
167  PRINT : PRINT "POSTING A REC
     ORD WILL:": PRINT : PRINT "
       - DELETE RECORD FROM ";F$;
     " FILE"
168  PRINT "   - ADD AMOUNT TO Y-
     T-D BALANCES"
169  PRINT "   - WRITE RECORD TO
     POSTED ";F$;" FILE": PRINT :
      PRINT
170  VTAB 15: PRINT "WHICH RECORD
      # DO YOU WANT TO POST?"
171  VTAB 16: CALL  - 868: INVERSE
     : INPUT "ENTER RECORD # OR T
     YPE 'END': ";DL$
172  IF DL$ = "END" THEN  NORMAL
     : GOTO 183
173 DL =  VAL (DL$):DL$ = "": IF
     DL > R THEN  VTAB 18: PRINT
     "NO SUCH RECORD. HIGHEST REC
     ORD IS ";R: GOTO 170
174  IF W$(DL,1) = " " AND W$(DL,
     2) = " " THEN  VTAB 18: PRINT
     "RECORD ALREADY POSTED": GOTO
     170
175  VTAB 18: CALL  - 868: PRINT
     "DELETED: ";W$(DL,6);"$";W$(
     DL,5)
176 R2 = R2 + 1
177  PRINT D$;"WRITE POSTED ";F$;
     ",R";R2
178  PRINT W$(DL,1): PRINT W$(DL,
     2): PRINT W$(DL,3): PRINT W$
     (DL,4): PRINT W$(DL,5): PRINT
     W$(DL,6)
179  PRINT D$
180 Z = Z + 1: FOR X = 1 TO 5:W(Z
     ,X) =  VAL (W$(DL,X)): NEXT
     X
181 W$(DL,1) = " ":W$(DL,2) = " "
182  GOTO 171
183  REM
184 N = 0:S = 0
185 S = S + 1
186 N = N + 1
187  IF N > R THEN 191
188  IF W$(N,1) = " " AND W$(N,2)
      = " " THEN 186
189  FOR X = 1 TO 6:W$(S,X) = W$(
     N,X): NEXT X
190  GOTO 185
191  REM
192 I = S - 1:R = 0:SW = 1: GOSUB
     121:SW = 0
193  REM
194  PRINT D$;"WRITE POSTED ";F$;
     ",RO"
195  PRINT R2
196  PRINT D$;"CLOSE POSTED ";F$
197  REM
198  ONERR  GOTO 220
199  TEXT : HOME : VTAB 10: FLASH
     : PRINT "READING YEAR-T-O-DA
     TE FILES": NORMAL
200  PRINT D$;"OPEN BALANCES"
201  PRINT D$;"READ BALANCES"
202  FOR X = 1 TO 24: FOR N = 1 TO
     12
203  INPUT BAL(X,N)
204  NEXT N,X
205  PRINT D$;"CLOSE BALANCES"
206  IF SW = 1 THEN  RETURN
207  REM
208  FOR X = 1 TO Z:BAL(W(X,1),W(
     X,3)) = BAL(W(X,1),W(X,3)) +
     W(X,5)
209  NEXT X
210  PRINT D$;"OPEN BALANCES"
211  PRINT D$;"WRITE BALANCES"
212  FOR X = 1 TO 24: FOR N = 1 TO
     12
213  PRINT BAL(X,N)
214  NEXT N,X
215  PRINT D$;"CLOSE BALANCES"
216  REM
217  IF PRTYPE = 1 THEN  GOSUB 38
     3
218  IF PRTYPE = 2 THEN SW = 1: GOSUB
     76:SW = 0
219  GOTO 527
220  REM
```

2.	BOOKS/MAGAZINES	$	442.00
3.	BUSINESS EXP.	$	246.00
4.	CLOTHES (ADULT)	$	244.00
5.	CLOTHES (CHILDREN)	$	123.00
6.	DONATIONS	$	0.00
7.	ELECTRICITY	$	0.00
8.	ENTERTAINMENT	$	0.00
9.	FOOD	$	0.00
10.	GIFTS	$	0.00
11.	HO[illegible]	$	0.00
12.	HO[illegible]REN)	$	0.00
13.	HOME [illegible]	$	0.00
14.	HOME MAINT	$	0.00
15.	INSURANCE	$	0.00
16.	INVESTMENTS	$	0.00
17.	LIQ-CIG'S	$	0.00
18.	MORTGAGE	$	0.00
19.	TELEPHONE	$	0.00
20.	TAX-INCOME	$	0.00
21.	TAX-COMPANY	$	0.00
22.	TAX-[illegible]	$	0.00
23.	VACATION	$	0.00

```
221 ER =  PEEK (222): POKE 216,0:
     IF ER <  > 5 THEN 141
222  PRINT D$;"CLOSE BALANCES"
223  PRINT D$;"OPENBALANCES"
224  PRINT D$;"WRITEBALANCES,RO"
225  PRINT 0
226  PRINT D$;"CLOSEBALANCES"
227  GOTO 207
228  REM
229 H = 0:K = 0:J = 0:V = 0:R = 0
    :CM = 0:F = 0
230  FLASH : PRINT "INITIALIZING
     SORT TABLE": NORMAL
231  FOR X = 0 TO 150: FOR YY = 0
      TO 5:WK(X,YY) = 0: NEXT : NEXT

232 SW = 1: GOSUB 143:SW = 0
233  CALL  - 936: HTAB 8: PRINT "
     * SORT ";F$;" RECORDS *
234  PRINT : PRINT "WHICH FIELD D
     O YOU WANT TO SORT ON?"
235  HTAB 10: PRINT "1 = ACCOUNT
     NUMBER": HTAB 10: PRINT "2 =
      CR CARD/CHECK NUMBER": HTAB
     10: PRINT "3 = MONTH ": HTAB
     10: PRINT "4 = DOLLAR AMOUNT
     "
236  PRINT : INVERSE : INPUT "ENT
     ER YOUR SELECTION: ";SR: NORMAL

237  IF SR = 1 THEN A$ = "BY ACCO
     UNT NUMBER"
238  IF SR = 2 THEN A$ = "BY CARD
     /CHECK NUMBER"
239  IF SR = 3 THEN A$ = "BY MONT
     H"
240  IF SR = 4 THEN A$ = "BY DOLL
     AR AMOUNT":SR = 5
241  IF SR > 5 THEN 233
242  FOR X = 1 TO R: FOR N = 1 TO
     5:WK(X,N) =  VAL (W$(X,N)): NEXT
     N,X
243  FLASH : PRINT "SORT IN PROSR
     ESS": NORMAL : GOSUB 19
244  FOR X = 1 TO R: FOR N = 1 TO
     5:W$(X,N) =  STR$ (WK(X,N)):
      NEXT N,X
245 X$ = "":I = R: PRINT "PRINT S
    ORTED LIST ON PRINTER? ": INVERSE
    : INPUT "ENTER 'Y' OR 'N' :"
    ;X$: NORMAL
246  IF X$ = "Y" AND PRTYPE = 1 THEN
      GOSUB 383
247  IF X$ = "Y" AND PRTYPE = 2 THEN
     SW = 1: GOSUB 76:SW = 0
248 R = 0
249  PRINT "SAVE SORTED LIST ON T
     HE DISK?": INVERSE : INPUT "
     ENTER 'Y' OR 'N' : ";X$: NORMAL
     : IF X$ = "Y" THEN SW = 1: GOSUB
     121:SW = 0
250  GOSUB 74
251  IF X$ = "Y" OR X$ = "N" THEN
     253
252  GOSUB 75: GOTO 249
253  GOTO 527
254  REM
255  TEXT : HOME
256  INVERSE : HTAB 12: PRINT "CO
     MPU-AGE 1984": NORMAL
257  HTAB 12: PRINT "** REPORT OP
     TIONS **"
258  PRINT : PRINT "1 = LIST AND
     TOTAL CREDIT CARD FILE"
259  PRINT "2 = LIST AND TOTAL CH
     ECK FILE"
260  PRINT "3 = TREND ANALYSIS &
     AVERAGE MONTH"
261  PRINT "4 = YEAR TO DATE SPEN
     DING PROFILE"
262  PRINT "5 = HI-RES SPENDING G
     RAPH"
263  PRINT "6 = LIST POSTED CREDI
     T CARD FILE"
264  PRINT "7 = LIST POSTED CHECK
      FILE"
265  VTAB 15: INVERSE : INPUT "EN
     TER SELECTION :";RP: NORMAL

266  GOSUB 74
267  IF RP = 1 THEN FL = 1:F$ = "
     CARD #": GOTO 275
268  IF RP = 2 THEN FL = 2:F$ = "
     CHECK #": GOTO 275
269  IF RP = 3 THEN  GOTO 282
270  IF RP = 4 THEN  GOTO 336
271  IF RP = 6 THEN FL = 1:F$ = "
     POSTED CARD #": GOTO 462
272  IF RP = 7 THEN FL = 2:F$ = "
     POSTED CHECK #": GOTO 462
273  IF RP <  > 5 THEN  GOSUB 75:
      GOTO 265
274  PRINT D$;"RUN PASG"
275  REM
276 T = 0: FOR X = 0 TO 5:CC(X) =
     0: NEXT X
277 SW = 1: GOSUB 143:SW = 0: REM
      ** READ FILE **
278 I = R: IF I = 0 THEN  VTAB 24
     : INVERSE : PRINT "NO RECORD
     S IN ";F$: NORMAL : FOR X =
     1 TO 1000: NEXT X: GOTO 281
279  IF PRTYPE = 1 THEN  GOSUB 38
     3
280  IF PRTYPE = 2 THEN SW = 1: GOSUB
     76:SW = 0
281  GOTO 527
282  REM
```

```
283 TA = 0:AV(0) = 0:TL(0) = 0: FOR
    X = 1 TO 12:TL(X) = 0:AV(X) =
    0:AV(X + 12) = 0: NEXT X
284 SW = 1: GOSUB 197
285 F$ = "CARD #": GOSUB 143
286  IF BZ = 1 THEN BZ = 0: GOTO
    288
287  GOSUB 375
288 F$ = "CHECK #": GOSUB 143
289  IF BZ = 1 THEN BZ = 0: GOTO
    291
290  GOSUB 375
291 SW = 0
292  FOR X = 1 TO 24: FOR N = 1 TO
    MT:AV(X) = AV(X) + BAL(X,N):
    TL(N) = TL(N) + BAL(X,N): NEXT
    N,X
293  FOR X = 1 TO 24:AV(X) = AV(X
    ) / MT:AV(X) =  INT (AV(X) *
    100 + .5) / 100:TA = TA + AV
    (X): IF TL(MT) > 0 THEN PC(X
    ,1) =  INT ((BAL(X,MT) * 100
    ) / TL(MT))
294  IF TL(MT - 1) > 0 THEN PC(X,
    2) =  INT ((BAL(X,MT - 1) *
    100) / TL(MT - 1))
295  NEXT X
296  FOR X = 1 TO 24: IF TA > 0 THEN
    PC(X,3) =  INT ((AV(X) * 100
    ) / TA)
297 NM$(X) = N$(X): NEXT X
298  IF PRTYPE = 1 THEN  GOTO 414
299  PRINT D$;"PR#1"
300  PRINT  CHR$ (9);"90N": IF PQ
    = 3 THEN  PRINT  CHR$ (15)
301  IF PQ = 1 THEN  PRINT  CHR$
    (2); CHR$ (30)
302  POKE 36,24: PRINT "TREND ANA
    LYSIS FOR THE MONTH OF ** ";
    MO$(MT);" **": PRINT
303 XM = 4:XP = 2
304  PRINT "ACCOUNT DESCR'
    ";: PRINT  TAB( 8)"CURR MO
    ";: PRINT  TAB( XP)"% TOT  "
    ;: PRINT  TAB( XM)"LAST MO
    ";: PRINT  TAB( XP)"% TOT  "
    ;: PRINT  TAB( XM)"AVG MO  "
    ;: PRINT  TAB( XP)"% TOT  ";
305  PRINT  TAB( XM)"CUR VS. AVG"
306  FOR X = 1 TO 80: PRINT "=";:
    NEXT X: PRINT
307  FOR X = 1 TO 24
308 PK = 1:V$ = NM$(X): GOSUB 11:
309  IF BAL(X,MT) = 0 THEN 314
310 P = BAL(X,MT): GOSUB 335:B =
    5
311 V$ =  STR$ (BAL(X,MT)):PK = (
    20 + B - C - 1): GOSUB 11
312 P = PC(X,1): GOSUB 335:B = 3
313 V$ =  STR$ (PC(X,1)) + "%":PK
    = (30 + B - C - 1): GOSUB 1
    1
314  IF BAL(X,MT - 1) = 0 THEN 31
    9
315 P = BAL(X,MT - 1): GOSUB 335:
    B = 5
316 V$ =  STR$ (BAL(X,MT - 1)):PK
    = (36 + B - C - 1): GOSUB 1
    1
317 P = PC(X,2): GOSUB 335:B = 3
318 V$ =  STR$ (PC(X,2)) + "%":PK
    = (46 + B - C - 1): GOSUB 1
    1
319  IF AV(X) = 0 THEN  PRINT : GOTO
    328
320 P = AV(X): GOSUB 335:B = 5
321 V$ =  STR$ (AV(X)):PK = (51 +
    B - C - 1): GOSUB 11
322 P = PC(X,3): GOSUB 335:B = 3
323 V$ =  STR$ (PC(X,3)) + "%":PK
    = (61 + B - C - 1): GOSUB 1
    1
324  IF BAL(X,MT) > AV(X) THEN X$
    = "ABOVE AVG"
325  IF BAL(X,MT) < AV(X) THEN X$
    = "BELOW AVG"
326  IF BAL(X,MT) = AV(X) THEN X$
    = "EQUALS AVG"
327 V$ = X$:PK = 70: GOSUB 15
328 PK = 0:SY = 0: NEXT X
329  FOR X = 1 TO 80: PRINT "-";:
    NEXT X: PRINT
330 PK = 1:V$ = "** TOTALS **": GOSUB
    11
331 P = TL(MT): GOSUB 335:B = 5:V
    $ =  STR$ (TL(MT)):PK = 20 +
    B - C - 1: GOSUB 11
332 P = TL(MT - 1): GOSUB 335:B =
    5:V$ =  STR$ (TL(MT - 1)):PK
    = 36 + B - C - 1: GOSUB 11
333 P = TA: GOSUB 335:B = 5:V$ =
    STR$ (TA):PK = 51 + B - C -
    1: GOSUB 15
334  GOTO 377
335 C = (P >  = 10) + (P >  = 100
    ) + (P >  = 1000) + (P >  =
    10000): RETURN
336  REM
337  FOR X = 1 TO 12:TL(X) = 0: NEXT
    X
338 SW = 1: GOSUB 197
339 F$ = "CARD #": GOSUB 143
340  REM
```

```
341  IF BZ = 1 THEN BZ = 0: GOTO
     343
342  GOSUB 375
343 F$ = "CHECK #": GOSUB 143
344  IF BZ = 1 THEN BZ = 0: GOTO
     346
345  GOSUB 375
346 SW = 0
347  IF PRTYPE = 1 THEN 443
348  PRINT  CHR$ (4);"PR#1"
349  PRINT  CHR$ (9);"130N"
350  IF PQ = 1 THEN  PRINT  CHR$
     (2); CHR$ (31)
351  IF PQ = 2 THEN  PRINT  CHR$
     (27); CHR$ (20)
352  IF PQ = 3 THEN  PRINT  CHR$
     (15)
353 PK = 30:V$ = "YEAR-TO-DATE SP
     ENDING ANALYSIS": GOSUB 15:P
     K = 35:V$ = "TODAY'S DATE: "
      +  STR$ (MT) + "/" +  STR$
     (D) + "/" +  STR$ (Y): GOSUB
     15
354 XN = 7:XJ = 10
355  PRINT  TAB( 1)"EXPENSE DESCR
     ";: PRINT  TAB( XJ)MO$(
     1);: PRINT "      ";: PRINT
      TAB( XN)MO$(2);: PRINT "
      ";: PRINT  TAB( XN)MO$(3
     );: PRINT "      ";: PRINT  TAB(
     XN)MO$(4);: PRINT "      ";
     : PRINT MO$(5);
356  PRINT "      ";: PRINT  TAB(
     XN)MO$(6);: PRINT "      ";
357  PRINT  TAB( XN)MO$(7);: PRINT
     "      ";: PRINT  TAB( XN)MO
     $(8);: PRINT "      ";: PRINT
      TAB( XN)MO$(9);: PRINT "
      ";: PRINT  TAB( XN)MO$(10
     );: PRINT "      ";: PRINT  TAB(
     XN)MO$(11);: PRINT "      ";: PRINT
      TAB( XN)MO$(12)
358  FOR X = 1 TO 130: PRINT "=";
     : NEXT X
359  PRINT :L = 0
360  FOR X = 1 TO 24:NM$(X) = N$(
     X): NEXT X
361  FOR X = 1 TO 24:PK = 1:V$ =
     NM$(X): GOSUB 11
362  FOR N = 1 TO 12: IF BAL(X,N)
      = 0 THEN 366
363 P = BAL(X,N): GOSUB 335
364 B = 14:PK = ((N * 9) + B - C -
     1):V$ =  STR$ (BAL(X,N)): GOSUB
     11
365 TL(N) = TL(N) + BAL(X,N)
366  NEXT N:PK = 0:SY = 0: PRINT
367  NEXT X
368  FOR X = 1 TO 130: PRINT "-";
     : NEXT X: PRINT
369 PK = 1:V$ = "** TOTALS ** ": GOSUB
     11
370  FOR X = 1 TO 12
371 P = TL(X): GOSUB 335
372 B = 14:PK = ((X * 9) + B - C -
     1):V$ =  STR$ (TL(X)): GOSUB
     11
373  NEXT X: PRINT
374  GOTO 378
375  FOR X = 1 TO R:WK(X,1) =  VAL
     (W$(X,1)):WK(X,2) =  VAL (W$
     (X,3)):WK(X,3) =  VAL (W$(X,
     5))
376 BAL(WK(X,1),WK(X,2)) = BAL(WK
     (X,1),WK(X,2)) + WK(X,3): NEXT
     X: RETURN
377  IF PQ = 3 THEN  PRINT  CHR$
     (18)
378  PRINT  CHR$ (9);"40N"
379  IF PQ = 2 THEN  PRINT  CHR$
     (27); CHR$ (19)
380  IF PQ = 3 THEN  PRINT  CHR$
     (18)
381  PRINT D$;"PR#0"
382  CLEAR :BU = 1: GOTO 487
383 X = 0:YY = T:T = 0: IF XX THEN
     T = YY
384  FOR J = 1 TO 5:CC(J) = 0: NEXT

385  HOME : PRINT  TAB( 7);F$;" S
     UMMARY REPORT"
386  IF OM = 3 THEN  PRINT  TAB(
     7)"SORTED ";A$
387  PRINT : PRINT "*  * PAID TO
     * A# ";
388  IF FL = 1 THEN  PRINT "CARD#
     ";
389  IF FL = 2 THEN  PRINT "CHECK
     #";
390  PRINT " MO DY AMOUNT": PRINT
     :CTR = 0
391  POKE 34, PEEK (37) + 1
392 X = X + 1: IF X > I THEN  GOTO
     405
393 CTR = CTR + 1: IF CTR > 16 THEN
      VTAB 24: INVERSE : INPUT "H
     IT RETURN TO CONTINUE";X$: NORMAL
     : VTAB 5: CALL  - 958:CTR =
     1
394  PRINT X;
395  POKE 36,3: PRINT  LEFT$ (W$(
     X,6),10);
396 T =  INT (T * 100 + .5) / 100
     :CC(CC) =  INT (CC(CC) * 100
     + .5) / 100
```

```
397  POKE 36,23 -  LEN (W$(X,2)):
      PRINT W$(X,2);
398  POKE 36,27 -  LEN (W$(X,3)):
      PRINT W$(X,3);
399  POKE 36,30 -  LEN (W$(X,4)):
      PRINT W$(X,4);
400 W =  VAL (W$(X,5))
401 T = T + W: IF FL = 1 THEN CC =
      VAL (W$(X,2)):CC(CC) = CC(C
     C) + W
402 P = W: GOSUB 108
403  POKE 36,27 + B - C - 1: PRINT
     W
404  GOTO 392
405  FOR X = 0 TO 38: PRINT "-";:
      NEXT X: PRINT
406 P = T: GOSUB 108
407  POKE 36,1: PRINT "*** TOTALS
      ***";: POKE 36,26 + B - C: PRINT
     T
408  IF FL <  > 1 THEN 412
409  VTAB 24: INVERSE : INPUT "HI
     T RETURN FOR CARD TOTALS ";X
     $: NORMAL
410  TEXT : HOME : HTAB 10: PRINT
     "* CARD TOTALS *"
411  VTAB 4: FOR X = 1 TO 5:P = C
     C(X): GOSUB 108: PRINT CD$(X
     );: POKE 36,(15 + B - C - 1)
     : PRINT CC(X): NEXT X
412  VTAB 24: INVERSE : INPUT "HI
     T RETURN TO CONTINUE";X$: NORMAL

413  TEXT : RETURN
414  HOME
415  POKE 36,4: PRINT "TREND ANAL
     YSIS FOR ** ";MO$(MT);" **"
416 XM = 4:XP = 2
417  PRINT "DSCR";: POKE 36,9: PRINT
     "CUR MO";: POKE 36,18: PRINT
     "%";: POKE 36,21: PRINT "LST
      MO";: POKE 36,29: PRINT "%"
     ;: POKE 36,32: PRINT "AVG MO
     "
418  FOR X = 0 TO 39: PRINT "=";:
      NEXT X: PRINT
419  FOR X = 1 TO 24
420  PRINT  LEFT$ (NM$(X),9);
421  IF BAL(X,MT) = 0 THEN 426
422 P = BAL(X,MT): GOSUB 442:B =
     5
423  POKE 36,8 + B - C - 1: PRINT
     BAL(X,MT);
424 P = PC(X,1): GOSUB 442:B = 3
425  POKE 36,(17 + B - C - 1): PRINT
     PC(X,1);
426  IF BAL(X,MT - 1) = 0 THEN 43
     1
427 PC = BAL(X,MT - 1): GOSUB 442:
     B = 5
428  POKE 36,(20 + B - C - 1): PRINT
     BAL(X,MT - 1);
429 P = PC(X,2): GOSUB 442:B = 3
430  POKE 36,(28 + B - C - 1): PRINT
     PC(X,2);
431  IF AV(X) = 0 THEN  PRINT : GOTO
     435
432 P = AV(X): GOSUB 442:B = 5
433  POKE 36,31 + B - C - 1: PRINT
     AV(X)
434 P = PC(X,3): GOSUB 442:B = 3
435  IF X = 18 THEN  INPUT "HIT R
     ETURN FOR LAST PART OF REPOR
     T";XX$
436  NEXT X
437  PRINT "** TOTALS **";
438 P = TL(MT): GOSUB 442:B = 5: POKE
     36,9 + B - C - 1: PRINT TL(M
     T);
439 P = TL(MT - 1): GOSUB 442:B =
     5: POKE 36,21 + B - C - 1: PRINT
     TL(MT - 1);
440 P = TA: GOSUB 442:B = 5: POKE
     36,32 + B - C - 1: PRINT TA
441  INPUT "PRESS RETURN TO GO TO
      MENU ";XX$: GOTO 461
442 C = (P >  = 10) + (P >  = 100
     ) + (P >  = 1000) + (P >  =
     10000): RETURN
443  REM

444  HOME :ZT = 8
445  POKE 36,5: PRINT "YEAR-TO-DA
     TE SPENDING ANALYSIS": POKE
     36,5: PRINT "TODAY'S DATE: "
     ;MT;"/";D;"/";Y
446  FOR N = 1 TO 12 STEP 3: PRINT
     "DESCR";
447  FOR I = N TO N + 2:ZT = ZT +
     9:: POKE 36,ZT: PRINT MO$(I)
     ;: NEXT I: PRINT :ZT = 8
448  FOR X = 1 TO 24: PRINT  LEFT$
     (N$(X),10);
449  FOR I = N TO N + 2:ZT = ZT +
     9:P = BAL(X,I): GOSUB 442
450  POKE 36,ZT - C - 1: PRINT BA
     L(X,I);:TL(I) = TL(I) + BAL(
     X,I): NEXT I: PRINT :ZT = 8
451  IF X = 18 THEN  INPUT "PRESS
      RETURN FOR REST OF REPORT";
     XX$
452  NEXT X
453  FOR YY = 0 TO 39: PRINT "-";
     : NEXT YY: PRINT
454  PRINT "**TOTALS**";:ZT = 8: FOR
     I = N TO N + 2:ZT = ZT + 9:P
```

```
2. BOOKS/MAGAZINES          $  442.00
3. BUSINESS EXP.            $  246.00
4. CLOTHES (ADULT)          $  244.00
5. CLOTHES (CHILDREN)       $  123.00
6. DONATIONS                $    0.00
7. ELECTRICITY              $    0.00
8. ENTERTAINMENT            $    0.00
9. FOOD                     $    0.00
10. GIFTS                   $    0.00
11. HOBBIES (ADULT)         $    0.00
12. HOBBIES (CHILDREN)      $    0.00
13. HOME-GAS                $    0.00
14. HOME-MAINT.             $    0.00
15. INSURANCE               $    0.00
16. INVESTMENT              $    0.00
17. LIQ-CIG'S               $    0.00
18. MORTGAGE                $    0.00
19. TELEPHONE               $    0.00
20. TAX-INCOME              $    0.00
21. TAX-COMPANY             $    0.00
22. TAX-INVEST              $    0.00
23. VACATION                $    0.00
```

```
     = TL(I): GOSUB 442: POKE 36
     ,ZT - C - 1: PRINT TL(I);: NEXT
     I: PRINT :ZT = 8
455  INPUT "PRESS RETURN FOR NEXT
      3 MONTHS";XX$
456  NEXT N
457  INPUT "PRESS RETURN TO GO TO
      MENU";XX$
458  GOTO 461
459  FOR X = 1 TO R:WK(X,1) =  VAL
     (W$(X,1)):WK(X,2) =  VAL (W$
     (X,3)):WK(X,3) =  VAL (W$(X,
     5))
460 BAL(WK(X,1),WK(X,2)) = BAL(WK
     (X,1),WK(X,2)) + WK(X,3): NEXT
     X: RETURN
461  GOTO 527
462  REM
463 T = 0:E1 = 0: FOR X = 0 TO 5:
     CC(X) = 0: NEXT X
464  ONERR  GOTO 482
465  PRINT D$;"OPEN ";F$;",L40"
466  PRINT D$;"READ ";F$;",R0"
467  INPUT R
468  IF R = 0 THEN  GOTO 483
469  REM
470 XX = 0
471  FLASH : PRINT "READ 100 RECO
     RDS OF ";F$: NORMAL
472 B1 = E1 + 1: IF (R - E1) > 10
     0 THEN E1 = E1 + 100:I = 100
     : GOTO 474
473  IF (R - E1) < = 100 THEN I =
     (R - E1):E1 = R
474  FOR Z = B1 TO E1:XX = XX + 1
475  PRINT D$;"READ ";F$;",R";Z
476  INPUT W$(XX,1),W$(XX,2),W$(X
     X,3),W$(XX,4),W$(XX,5),W$(XX
     ,6)
477  NEXT
478  PRINT D$: IF PRTYPE = 1 THEN
     SW = 1: GOSUB 383:SW = 0
479  IF PRTYPE = 2 THEN SW = 1: GOSUB
     76:SW = 0
480  IF E1 < R THEN  GOTO 469
481  GOTO 483
482 ER =  PEEK (222): POKE 216,0:
      IF ER <  > 5 THEN 141
483  PRINT D$;"CLOSE ";F$
484  PRINT D$: IF E1 = 0 THEN  INVERSE
     : PRINT "NO RECORDS IN ";F$:
      NORMAL : FOR X = 1 TO 1000:
      NEXT X
485 XX = 0: GOTO 527
486  REM
487  TEXT : HOME
488 XL =  PEEK (54):XH =  PEEK (5
     5)
489  HTAB 7: PRINT "HOME FINANCE
     MANAGEMENT"
490  FOR I = 0 TO 39: PRINT "=";:
      NEXT I
491  PRINT "THIS IS YOUR PERSONAL
      SYSTEM FOR MANAG-": PRINT "
     ING CREDIT CARDS, CHECKS AND
      OVERALL": PRINT "HOME FINAN
     CES."
492  PRINT : PRINT "YOU'LL BE ASK
     ED TO SELECT FROM THESE": PRINT
     "OPTIONS:"
493  HTAB 10: PRINT "ADD": HTAB 1
     0: PRINT "POST": HTAB 10: PRINT
     "SORT": HTAB 10: PRINT "REPO
     RT": HTAB 10: PRINT "END THE
      SESSION"
494  PRINT : PRINT " PAS WILL MAI
     NTAIN YOUR EXPENSE DATA": PRINT
     "IN THE FOLLOEING FILES:"
495  PRINT : HTAB 10: PRINT "CRED
     IT CARD FILE": HTAB 10: PRINT
     "CHECK FILE": HTAB 10: PRINT
     "YEAR-TO-DATE FILE"
496  HTAB 10: PRINT "POSTED CREDI
     T CARD FILE": HTAB 10: PRINT
     "POSTED CHECK FILE"
497  VTAB 24: INVERSE : INPUT "HI
     T RETURN TO CONTINUE ";X$: NORMAL
498  HOME : POKE 33,37: POKE 32,3
499  PRINT : HTAB 13: PRINT "* AD
     D *"
500  PRINT : PRINT "USE THE 'ADD'
      OPTION TO ENTER DETAIL": PRINT
     "EXPENSE TRANSACTIONS INTO T
     HE CREDIT": PRINT "CARD AND
     CHECK FILES."
501  VTAB 8: HTAB 13: PRINT "* PO
     ST *"
502  PRINT : PRINT "USE THE 'POST
     ' OPTION TO REMOVE": PRINT "
     A DETAIL TRANSACTION FROM IT
     S": PRINT "ACTIVE FILE AND U
     PDATE THE YEAR-": PRINT "TO-
     DATE BALANCES. NOTE: EACH"
503  PRINT "'POSTED' TRANSACTION
     WILL BE WRITTEN": PRINT "TO
     AN AUDIT TRAIL FILE FOR FUTU
     RE": PRINT "REFERENCE."
504  PRINT : HTAB 13: PRINT "*  *
      *"
505  PRINT : PRINT "FOR EACH OPTI
     ON, FOLLOW THE ": PRINT "INS
```

```
     TRUCTIONS ON THE SCREEN."
506  VTAB 24: INVERSE : INPUT "HI
     T RETURN TO CONTINUE ";X$: NORMAL
     : POKE 32,0: POKE 33,40
507  REM
508  TEXT : HOME
509  HTAB 5: PRINT "PERSONAL ACCO
     UNTING SYSTEM": HTAB 12: INVERSE
     : PRINT "COMPU-AGE 1984": NORMAL
510  HTAB 8: PRINT "* INTIALIZATI
     ON SCREEN *"
511  VTAB 3: PRINT "ENTER THE REP
     ORT OUTPUT DEVICE": PRINT "Y
     OU WISH TO USE."
512  PRINT : HTAB 10: PRINT "1 =
     SCREEN ONLY": HTAB 10: PRINT
     "2 = PRINTER"
513  VTAB 10: INVERSE : INPUT "EN
     TER YOUR SELECTION :";PRTYPE
     : NORMAL
514  IF PRTYPE < 1 OR PRTYPE > 2 THEN
      GOSUB 75: GOTO 513
515  IF PRTYPE = 2 THEN  PRINT : PRINT
     "1= CENTRONICS COMPATIBLE": PRINT
     "2= CONTRONICS 737": PRINT "
     3= EPSON MX-80 PRINTER": INPUT
     "ENTER SELECTION ";PQ: IF PQ
      < 1 OR PQ > 3 THEN 515
516  REM
517  VTAB 20: INVERSE : PRINT "EN
     TER THE REPORT HEADING DATE:
      ": NORMAL
518  VTAB 22: CALL  - 868: INVERSE
     : INPUT "IN MO,DAY,YR:";MT,D
     ,Y: NORMAL
519  IF MT > 12 OR D > 31 THEN  GOSUB
     75: GOTO 518
520  REM
521 D$ =  CHR$ (4)
522  DIM N$(24),MO$(13),CD$(5)
523  FOR X = 1 TO 24: READ N$(X):
      NEXT X
524  FOR X = 1 TO 12: READ MO$(X)
     : NEXT X
525  FOR X = 1 TO 5: READ CD$(X):
      NEXT X
526  DIM BAL(24,12),W$(150,6),W(7
     5,13),TL(12),NM$(24),AV(24),
     PC(24,3),WK(150,5),CC(5)
527  REM
528  CALL  - 936
529 I = 0
530  INVERSE : HTAB 12: PRINT "CO
     MPU-AGE 1984": NORMAL
531  HTAB 13: PRINT "* MAIN MENU
     *"
532  VTAB 3: PRINT "ENTER THE NUM
     BER OF THE OPTION YOU WANT ":
     PRINT "TO USE.": HTAB 10: PRINT
     "1 = ADD RECORDS": HTAB 10: PRINT
     "2 = POST (DELETE) RECORDS"
533  HTAB 10: PRINT "3 = SORT REC
     ORDS ": HTAB 10: PRINT "4 =
     REPORT/LIST OPTIONS": HTAB 1
     0: PRINT "5 = END THE SESSIO
     N"
534  VTAB 11: INVERSE : INPUT "EN
     TER YOUR SELECTION: ";OM: NORMAL
535  GOSUB 74
536  IF OM = 5 THEN  PRINT "ALL D
     ONE": END
537  IF OM < 1 OR OM > 5 THEN  GOSUB
     75: GOTO 534
538  IF OM = 4 THEN  GOTO 545
539  VTAB 15: PRINT "ENTER THE NU
     MBER OF THE FILE TO BE USED.
     "
540  HTAB 10: PRINT "1 = CREDIT C
     ARDS": HTAB 10: PRINT "2 = C
     HECKS"
541  VTAB 20: INVERSE : INPUT "EN
     TER YOUR SELECTION :";FL: NORMAL
542  IF FL <  > 1 AND FL <  > 2 THEN
      GOSUB 75: GOTO 541
543  IF FL = 1 THEN F$ = "CARD #"
544  IF FL = 2 THEN F$ = "CHECK #
     "
545  ON OM GOTO 113,143,228,254
546  DATA    1. AUTO EXPENSE,2. BO
     OKS/MAGS.,3. BUSINESS EXP,4.
      CLOTHES ADULT,5. CLOTHES KI
     DS,6. DONATIONS,7. ELECTRICI
     TY,8. ENTERTAINMENT
547  DATA    9. FOOD,10. GIFTS,11.
     HOBBY-ADULT,12. HOBBY-KIDS,
     13. HOME-GAS,14. HOME-MAINT'
     ,15. INSURANCE,16. INVESTMEN
     TS,17. LIQ/CIG'S,18. MORTGAG
     E
548  DATA    19.TELEPHONE,20.TAX-I
     NCOME,21.TAX-COMPANY,22.TAX-
     INVEST.,23.VACATION,24.WATER
549  DATA    JAN,FEB,MAR,APR,MAY,J
     UN,JLY,AUG,SEP,OCT,NOV,DEC
550  DATA    MASTERCHARGE = $,VISA
     CARD    = $,AM EXPRESS = $,G
     ASOLINE    = $,OTHER CARDS =
     $
```

```
2.  BOOKS/MAGAZINES        $  442.00
3.  BUSINESS EXP.          $  246.00
4.  CLOTHES (ADULT)        $  244.00
5.  CLOTHES (CHILDREN)     $  123.00
6.  DONATIONS              $    0.00
7.  ELECTRICITY            $    0.00
8.  ENTERTAINMENT          $    0.00
9.  FOOD                   $    0.00
10. GIFTS                  $    0.00
11. HOBBIES (ADULT)        $    0.00
12. HOBBIES (CHILDREN)     $    0.00
13.HOME-GAS                $    0.00
14.HOME-MAINT              $    0.00
15.INSURANCE               $    0.00
16.INVESTMENTS             $    0.00
17.LIQ-CIG'S               $    0.00
18.MORTGAGE                $    0.00
19.TELEPHONE               $    0.00
20.TAX-INCOME              $    0.00
21.TAX-COMPANY             $    0.00
22.TAX-INVEST.             $    0.00
23.VACATION                $    0.00
```

```
2  REM  *********************
3  REM  *
4  REM  * PASG-
5  REM  * COMPU-AGE 1984
6  REM  *
7  REM  *********************
12  REM
13 D$ =  CHR$ (4)
14  PRINT D$"MON CIO"
15  ONERR  GOTO 116
16  PRINT D$;"BLOAD PASS"
17  HOME
18  VTAB 3: PRINT "********* SPEND
    ING PROFILE GRAPH *********"
19  PRINT : PRINT "CONTROLS:": PRINT
     TAB( 5)"ESC = DISPLAY TEXT
    SPENDING": PRINT  TAB( 5)"SP
    C = DISPLAY GRAPHICS": PRINT
     TAB( 5)"RTN = RETURN TO TRA
    C MENU"
20  PRINT
21 D$ =  CHR$ (4)
22  POKE 232,0: POKE 233,145
23  DIM A(24),B(24),DR$(20),VLOC(
    20),BAL(24,12)
24  DIM WK(150,5),W$(100,6),W(24,
    13),A$(24),T(24)
25  GOSUB 97
26  FOR I = 1 TO 24:B(I) = B(I) *
    (159 / 2500):T(I) = B(I): NEXT
27  REM  DRAW GRAPH
28  HGR2
29  ROT= 0: SCALE= 1
30  FOR I = 1 TO 20:AM = AM + 125
    :DR$(I) =  STR$ (AM): IF  LEN
    (DR$(I)) = 3 THEN DR$(I) =
    " " + DR$(I)
31  NEXT
32 J = 160
33  FOR I = 1 TO 20:J = J - 8:VLO
    C(I) = J: NEXT
34 J = 0
35  HCOLOR= 3: HPLOT 32,0 TO 32,1
    59 TO 279,159
36  FOR I = 152 TO 8 STEP  - 8: HPLOT
    32,I TO 34,I: NEXT
37  GOSUB 63
38  FOR I = 1 TO 24:A(I) = 42 + J
    :J = J + 9: NEXT
39  FOR I = 1 TO 24: HPLOT A(I),1
    57 TO A(I),158: NEXT
40  GOSUB 75
41 A(0) = 32:B(0) = 0
42  FOR I = 1 TO 24
43  IF 159 - B(I) < 0 THEN B(I) =
    159: DRAW 27 AT (A(I) - 3),0
    : CALL  - 198: GOTO 45
44  DRAW 11 AT (A(I) - 2),159 - B
    (I)
45  NEXT
46  FOR I = 0 TO 23: HPLOT A(I),1
    59 - B(I) TO A(I + 1),159 -
    (B(I + 1)): NEXT
47  REM  DISPLAY TEXT
48  HOME
49  FOR I = 1 TO 24: READ A$(I): NEXT
50  DATA  1. AUTO,2. BOOKS/MAGAZI
    NES,3. BUSINESS EXP.,4. CLOT
    HES (ADULT),5. CLOTHES (CHIL
    DREN),6. DONATIONS,7. ELECTR
    ICITY,8. ENTERTAINMENT,9. FO
    OD,10. GIFTS,11. HOBBIES (AD
    ULT),12. HOBBIES (CHILDREN)
51  DATA  13.HOME-GAS,14.HOME-MAI
    NT',15.INSURANCE,16.INVESTME
    NTS,17.LIQ-CIG'S,18.MORTGAGE
    ,19.TELEPHONE,20.TAX-INCOME,
    21.TAX-COMPANY,22.TAX-INVEST
    .,23.VACATION,24.WATER
52  HOME : FOR I = 1 TO 24: IF  VAL
    (A$(I)) < 10 THEN T = 2: GOTO
    55
53 T = 1
54  IF  VAL (A$(I)) = 24 THEN  PRINT
     TAB( T)A$(I);: HTAB 29: PRINT
    "$";:X$ =  STR$ (T(I) / (159
    / 2500)): GOSUB 109: HTAB (
    38 -  LEN (X$)): PRINT X$;: GOTO
    58
55  PRINT  TAB( T)A$(I);: HTAB 29
    : PRINT "$";:X$ =  STR$ (T(I
    ) / (159 / 2500)): GOSUB 109
    : HTAB (38 -  LEN (X$)): PRINT
    X$
56  NEXT
57  REM  KEYBOARD CONTROLS
58 X =  PEEK ( - 16384): IF X = 1
    55 THEN  POKE  - 16368,0: POKE
    - 16303,0: POKE  - 16300,0:
    POKE  - 16298,0: GOTO 58
59  IF X = 141 THEN  POKE  - 1636
```

```
   8,0: PRINT : PRINT  CHR$ (4)
   "RUN PAS"
60  IF X = 160 THEN  POKE  - 1636
   8,0: POKE  - 16297,0: POKE  -
   16304,0: POKE  - 16302,0: POKE
    - 16299,0
61  GOTO 58
62  REM  GRAPH NUMBERS
63  FOR I = 1 TO 20:HL = 0
64  FOR J = 1 TO  LEN (DR$(I))
65 D$ =  MID$ (DR$(I),J,1)
66  IF D$ = " " THEN 70
67 D =  VAL (D$)
68  IF D = 0 THEN D = 10
69  DRAW D AT HL,VLOC(I)
70 HL = HL + 5
71  NEXT : DRAW 12 AT HL,VLOC(I)
72  NEXT
73  RETURN
74  REM  CATEGORIES
75 DR$(1) = "ABBCCDEEFGHHGMIILMTT
   TTVW":DR$(2) = "UOULLOLNOIOO
   AANNQTEAAAAA":DR$(3) = "TOSO
   ONETOFBBSISVCGLXXXCT":DR$(4)
    = "OKITTACEDTBB N SGEEICIAR
   "
76 VL = 42
77 HL = 160: FOR I = 1 TO 4
78 HL = HL + 5
79  FOR J = 1 TO  LEN (DR$(I))
80 D$ =  MID$ (DR$(I),J,1)
81  IF D$ = " " THEN VL = VL + 9:
    GOTO 84
82 D =  ASC (D$):D = D - 52
83  DRAW D AT VL,HL:VL = VL + 9
84  NEXT :HL = HL + 2:VL = 42: NEXT
   : RETURN
85  REM  READ FILES
86  FLASH : PRINT "READING ";F$;"
    FILE": NORMAL
87  PRINT D$;"OPEN";F$;",L40"
88  PRINT D$;"READ";F$;",R0"
89  INPUT R
90  IF  NOT R THEN 95
91  FOR X = 1 TO R
92  PRINT D$;"READ";F$;",R";X
93  INPUT W$(X,1),W$(X,2),W$(X,3)
   ,W$(X,4),W$(X,5),W$(X,6)
94  NEXT X
95  PRINT D$;"CLOSE";F$
96  RETURN
97  FLASH : PRINT "READING YEAR-T
    O-DATE FILES": NORMAL
98  PRINT D$;"OPEN BALANCES"
99  PRINT D$;"READ BALANCES"
100  FOR X = 1 TO 24: FOR N = 1 TO
    12
101  INPUT BAL(X,N)
102  NEXT N,X
103  PRINT D$;"CLOSE BALANCES"
104 F$ = "CARD #": GOSUB 85: GOSUB
    108
105 F$ = "CHECK #": GOSUB 85: GOSUB
    108
106  FOR X = 1 TO 24: FOR N = 1 TO
    12:B(X) = B(X) + BAL(X,N): NEXT
    N,X
107  RETURN
108  FOR X = 1 TO R:WK(X,1) =  VAL
    (W$(X,1)):WK(X,2) =  VAL  (W$
    (X,3)):WK(X,3) =  VAL  (W$(X,
    5)):BAL(WK(X,1),WK(X,2)) = B
    AL(WK(X,1),WK(X,2)) + WK(X,3
    ): NEXT : RETURN
109  FOR Z = 1 TO  LEN (X$)
110  IF  MID$ (X$,Z,1) = "." THEN
    113
111  NEXT
112 X$ = X$ + ".00": RETURN
113  IF Z =  LEN (X$) - 2 THEN  RETURN

114  IF Z =  LEN (X$) - 1 THEN X$
    = X$ + "0": RETURN
115 X$ =  LEFT$ (X$,Z + 2): RETURN

116  PRINT : PRINT D$"CLOSE":ER =
    PEEK (222)
117  TEXT : HOME : VTAB 12
118  IF ER = 6 THEN  PRINT "FILE
    <";F$;"> DOES NOT EXIST.": GOTO
    121
119  IF ER = 8 THEN  PRINT "BAD D
    ISK OR DRIVE ERROR.": GOTO 1
    21
120  IF ER <  > 6 AND ER <  > 8 THEN
     PRINT "ERROR NO.";ER;" OCCU
    RRED IN LINE ";: PRINT  PEEK
    (218) +  PEEK (219) * 256
121  END
```

```
PASS

*9100.92D6

9100- 26 00 4E 00 54 00 5C 00
9108- 65 00 6C 00 76 00 80 00
9110- 86 00 90 00 9A 00 A2 00
9118- AD 00 BE 00 CB 00 D8 00
9120- E4 00 EE 00 F9 00 04 01
9128- 11 01 1D 01 27 01 31 01
9130- 3C 01 43 01 4F 01 5B 01
9138- 66 01 71 01 7F 01 8C 01
9140- 98 01 9F 01 A8 01 B2 01
9148- BD 01 C9 01 D3 01 09 3E
9150- 0E 36 06 00 29 2D BE BF
9158- 2E 2D 05 00 29 2D 36 3F
9160- 4E 3E 3F 07 00 31 2E 0C
9168- 2E 1E 36 00 29 2D DE 33
9170- 2D 15 1E 3F 07 00 29 2D
9178- DE 33 36 2D 25 3C 27 00
9180- 29 2D BE 17 36 00 09 AD
9188- 1E 0E 1E 3F 60 1C 07 00
9190- 29 2D 36 E7 33 75 31 3F
9198- 3F 00 09 AD 36 1E 3F 20
91A0- 3C 00 09 15 3F 17 2D 2D
91A8- 1E 3F 0E 05 00 91 12 05
91B0- 80 20 0C AD F6 77 09 48
91B8- 2D 20 1C BF 2E 00 92 52
91C0- 24 64 0C 15 0E 36 3E 80
91C8- 38 27 00 31 36 36 2D 2D
91D0- 20 1C 3F 4C E1 3F 06 00
91D8- 0A 0C 2D 15 DF 1A 36 0E
91E0- 2D 05 20 00 29 2D 15 36
91E8- F6 3F 27 24 3C 00 29 2D
91F0- B5 3B E7 36 36 2D 2D 04
91F8- 00 29 2D B5 3B E7 36 36
9200- 06 00 00 0A 09 2D 15 DE
9208- 2A 35 1E 3F 07 20 24 05
9210- 00 36 36 2E 80 80 28 25
9218- 08 36 36 26 00 2D 2D DE
9220- 36 FE 2A 2D 25 00 00 92
9228- 0A 0E 2D 20 24 07 28 2D
9230- 00 92 52 24 24 6C F1 1E
9238- 76 0E 06 00 31 36 36 2D
9240- 2D 05 00 92 52 24 24 AC
9248- 15 05 60 36 36 36 00 92
9250- 52 24 24 AC 15 15 56 24
9258- 24 3C 00 92 0A 24 64 2D
9260- 15 36 F6 3F 07 00 92 52
9268- 24 24 2C 2D 15 F6 3F 04
9270- 00 92 0A 24 64 2D 15 FE
9278- 0E 35 15 1F 3F 07 00 92
9280- 52 24 24 2C 2D 15 F6 3F
9288- 0E 15 06 00 0A 0C 2D F5
9290- 9B 2D 15 F6 3F 07 20 00
9298- 29 2D F5 33 36 36 00 31
92A0- 36 76 2D 05 20 24 3C 00
92A8- 31 36 0E 0E 05 28 20 24
92B0- 05 00 31 36 36 05 60 15
92B8- 0E 24 24 3C 00 31 0E BE
92C0- 6E 09 E4 07 28 05 20 07
92C8- 00 31 0E 05 40 31 1E 1E
92D0- 36 06 00 29 2D 15 4C
```

APPLESOFT程式語法偵錯工具

幫助你快速檢出打字錯誤

對於打字經常上有錯的電腦迷來說，本程式是不可缺少的工具。PSAS是PROGRAM SYNTAX ANAL-XSIS SYSTEM的簡稱，從字面上即可看出，它是APPLESOFT語法上的偵錯工具，一切有效的命令語法已包括在程式內，爲數約爲二百多個，而且它包括了符號（包括標點在內）的偵錯功能，因此在冗長程式內的程式偵測過程可以減低所耗費的精力，尤以偵測標點時爲甚，我相信大家都會明白箇中滋味的！

■ PSAS 的固件要求

1、48 K APPLE 及 AUTO ROM。

2、一部或以上的磁碟驅動器。

3、足夠所須的磁碟空間，因爲在偵測過程內，程式會寫出一些二進位及文件檔案以對應 CHECKCODE。在工作磁碟上需要至少保留約爲二倍代測程式長度的空間，否則會出現DOS的磁碟滿載訊號。

4、DOS 3.3 系統。

■ 使用方法

使用本程式相當簡單，在單磁碟機的使用下，只須先將待偵測的APPL-ESOFT 程式和本程式錄在同一面工作磁碟上，再按入 RUN PSAS，如圖一所示。

SOURCE FILE 爲待偵測的程式。

CHECK CODE FILE 按入 RETURN。

MEMORY USAGE 按 RET-URN。

OPTIONS 按 RETURN，若希望將偵測過程列印出，可以按入CTRL-D（即 CONTROL D）PR#1便可以了。

若使用兩部磁碟機時則在 SOURC-E FILE 上鍵入待測試的 FILE 名及其所在的磁碟機中，如 TEST，D2。

CHECK CODE FILE 上按入 FILE 名及所在之磁碟機如上，例如 TEST.OBJ，D2。

MEMORY USAGE 上按 RE-TURN。

OPTION 上可照前述去做便可以了。

本程式執行後將在寫入該程式的 CHECK CODE FILE，及一個名爲 CLINUM 的 FILE，讀者可以刪去它，省却磁碟空間。

本程式只能檢查以 APPLESOFT 寫出的程式，對於機械語言則無能爲力了！

本程式在程式磁碟上的 FILENA-ME 包括以下5個。

1、PSAS
2、CHECK CODE
3、PSAS 0
4、PSAS 1
5、PSAS 2

■

```
SOURCE FILE? TEST


CHECK  CODE FILE:
(DEFAULT TEST.OBJ)?


MEMORY USAGE:
DEFAULT CONFIGURATION?


OPTIONS:
DEFAULT CONFIGURATION?

*****BEGINNING PASS 1

10  PRINT "THIS IS A TEST"

CHECKING COMPLETE !

A CONTRIBUTION OF COMPUTING-AGE 1984
```

圖（1）

由於本檢錯程式長度相當長，超過150個 SECTOR，而且全部是機械語言，假如要全部刊出，會佔去約 15 頁版位，因此我們決定不將程式登錄在雜誌上，以節省寶貴的篇幅，而將整個程式錄在由 COMPUHOUSE 製作的今期程式磁碟上，讀者請留意。

數基轉換器

在 APPLE 的程式編寫編輯過程中，甚至很多工具程式裡都經常需要計算一些位址的十進制與十六進制換算，雖然以往會發表過很多這種功能的程式，但現在請閣下對以下所列舉的事實作出考慮，便會確信此一極限程式是所有這類程式中最好的。

①它的編寫方式是採用機械語言，因此運算速度很快。

②它的程式長度比一切的程式短，這樣便帶來：

(A) 運算速度更高（事實上執行的時間僅爲鍵入數值而已。）

(B) 佔用非常少的記憶空間。可以容許同時將 BASIC／滙編語言或二者同時載入記憶體中以作編輯或運行，而程式間的位址衝突情況可降至最低程度。

③在使用非常便利，僅需鍵入& 鍵而已。

□ 程式的載入步驟：

①在鍵入本文所附的兩個程式後，先存放在磁碟中，或從 COMPUHOUSE所提供的磁碟中載入記憶體裡，其中的 FILE 名爲 NUMBER CONVERTER MAKER，它的功能是產生一個 RANDOM ACCESS 的 TEXT FILE 。第二個機械語言主程式的 FILE 名爲 AMP NUMBER CONV。

□程式的使用方法：

(A) RUN NUMBER CONVERTER MAKER

(B) EXEC NUMBER CONVERTER

(C) 按入& ×××××

其中（C）的& ×××××爲標準輸入格式，& 是操作指令，×××××可以爲十六進制數字，亦可以爲正或負值的十進制數字，例如：

] & 53248(鍵入之十進制數字) $D000 53248 —12288 （顯示的結果數值）

] & —151

$FF69 65385 —151

□顯示格式：

由左至右起：

①爲相對應之十六進制數值

②爲相對應之十進制數值

③爲相對應之十進制負數值。（僅出示於鍵入數值大於 $7FFF時）

□ 應用實例：

任何所鍵入之數值／運算結果之數值不大於 65535 時，則可運算十進制的多項或換算，例如

(A)] & 4 ＊ 4096 （輸入 4 乘以 4096）

$4000 16384 （執行後的結果）

(B)] & 450.95—327.15（輸入十進制浮點小數加減）

$ 007B 123

NUMBER CONVERTER MAKER

```
100  REM  NUMBER CONVERTER MAKER
     -- MAKES AN EXEC FILE NUMBER
      CONVERTER TO DISPLAY INSTRU
     CTIONS AND RUN THE AMPERSAND
      NUMBER CONVERTER BINARY PRO
     GRAM
110  REM  TO USE, ENTER 'EXEC NUM
     BER CONVERTER'
120  PRINT  CHR$ (4);"OPEN NUMBER
      CONVERTER"
130  PRINT  CHR$ (4);"WRITE NUMBE
     R CONVERTER"
140  PRINT "PRINT" CHR$ (34)"TO C
     ONVERT NUMBERS, ENTER HEX OR
      DECIMALNUMBERS AS &$ED24, &
     60708, OR &-4828" CHR$ (34)
150  PRINT "BRUN AMPERSAND NUMBER
      CONVERTER": REM      LOADS F
     ROM $0345 TO $03CA
160  PRINT  CHR$ (4);"CLOSE NUMBE
     R CONVERTER"
170  END
```

□

數基轉換器

AMPERSAND NUMBER CONVERTER

```
0345-   A9 4C      LDA   #$4C
0347-   8D F5 03   STA   $03F5
034A-   A9 55      LDA   #$55
034C-   8D F6 03   STA   $03F6
034F-   A9 03      LDA   #$03
0351-   8D F7 03   STA   $03F7
0354-   60         RTS
0355-   C9 24      CMP   #$24
0357-   D0 35      BNE   $038E
0359-   A9 00      LDA   #$00
035B-   85 50      STA   $50
035D-   85 51      STA   $51
035F-   A0 02      LDY   #$02
0361-   B9 00 02   LDA   $0200,Y
0364-   F0 22      BEQ   $0388
0366-   C8         INY
0367-   49 30      EOR   #$30
0369-   C9 0A      CMP   #$0A
036B-   90 09      BCC   $0376
036D-   69 88      ADC   #$88
036F-   C9 FA      CMP   #$FA
0371-   B0 03      BCS   $0376
0373-   4C 94 03   JMP   $0394
0376-   A2 03      LDX   #$03
0378-   0A         ASL
0379-   0A         ASL
037A-   0A         ASL
037B-   0A         ASL
037C-   0A         ASL
037D-   26 50      ROL   $50
037F-   26 51      ROL   $51
0381-   CA         DEX
0382-   10 F8      BPL   $037C
0384-   C0 06      CPY   #$06
0386-   90 D9      BCC   $0361
0388-   20 95 D9   JSR   $D995
038B-   4C 94 03   JMP   $0394
038E-   20 7B DD   JSR   $DD7B
0391-   20 52 E7   JSR   $E752
0394-   A9 A4      LDA   #$A4
0396-   20 ED FD   JSR   $FDED
0399-   A6 50      LDX   $50
039B-   A5 51      LDA   $51
039D-   20 41 F9   JSR   $F941
03A0-   20 48 F9   JSR   $F948
03A3-   A6 50      LDX   $50
03A5-   A5 51      LDA   $51
03A7-   20 24 ED   JSR   $ED24
03AA-   20 48 F9   JSR   $F948
03AD-   A5 51      LDA   $51
03AF-   10 16      BPL   $03C7
03B1-   A9 AD      LDA   #$AD
03B3-   20 ED FD   JSR   $FDED
03B6-   18         CLC
03B7-   A5 50      LDA   $50
03B9-   49 FF      EOR   #$FF
03BB-   69 01      ADC   #$01
03BD-   AA         TAX
03BE-   A5 51      LDA   $51
03C0-   49 FF      EOR   #$FF
03C2-   69 00      ADC   #$00
03C4-   20 24 ED   JSR   $ED24
03C7-   20 BE FD   JSR   $FDBE
03CA-   60         RTS
```

美國最大的電腦市場研究公司 FU—TURE COMPUTING INC. 最近發表一份國際微型電腦市場統計報告，指出一九八三年全球的家庭電腦硬件零售總值爲二十六億美元，而美國市場佔其中百份之七十六。預計未來五年的發展會非常迅速，至一九八八年微型電腦市場的零售總值會增至九十四億美元，增幅達百份之三百六十。銷售額最高的電腦制造廠是美國COMMODORE確達電腦公司，在一九八三年度銷出微型電腦價值首次突破十億美元，佔整個市場約百份之三十八。

至於商用私人電腦市場方面，一九八三年零售總值爲七十三億美元，預測到一九八八年會增至三百零五億美元。

道歉

第五期「私人約會備忘日曆」程式列表行號5470 有錯誤，應改爲：

```
5470  IF T1$(I) = "99"
 THEN T1$(I ) = " ":T2$
 (I) = " ":T3$(I) =
 " ": GOTO 5520
```

此外同期「濃縮圖片儲存空間」程式漏登一段將於第七期更正

一九八四年青年科學家選舉

由九龍青年商會主辦的一九八四年青年科學家選舉現正開始接受報名。

是次比賽的宗旨是在於鼓勵青年人發揮對社會有貢獻之科學潛能。比賽將設冠、亞、季軍，藉以表揚以"美好家居在科技"爲主題所創作之優勝作品。

所有參賽者必須以個人或小組（每組不可多於 3 人）名義參加。參賽作品必須附有中／英文註明作品之特點及功能兩份。所有參賽作品須於五月十一日之前呈交：

九龍青年商會，青年科學家選舉小組主席收。

香港皇后大道中 272 號中央大廈15字樓C座。

入選者將被安排於五月廿日及六月廿四日會見評判委員會。

參加者年齡須於15至24歲之間。

（欲查詢有關詳情，可向九龍青年商會查詢）

GO][一個取代HELLO的程式

THOMAS LUK

相信各位讀者也有以下的經驗，就是當磁碟內的程式太多時（超過16個，如本刊的磁碟就是），當打入 CATA－LOG 時，程式的檔案名稱便會佔據了整個篇幅，更甚者，就是一些在最前排的檔案名字會在螢幕上消失，這樣當我們想執行那些程式時，便要重覆做一次，耗去不少時間。或者當我們想 RENAME 那些 LOCKED FILE 時，我們要先打入“UNLOCK XXX”，再打入“RENAME XXX，YYY”，然後又再打入“LOCK YYY”，這期間我們大約要打上好幾十個按鍵；還有一點要注意的，就是這幾十個按鍵中沒有一個按錯才能順利地完成，試問要重覆做這個動作是一件多惱人的事。

程式的功能

這個程式簡化了以上的步驟，而且還提供了不少功能予用家，如：RUN，LOAD，BRUN，BLOAD，EXEC，LOCK，UNLOCK，RENAME，DELETE，VER－IFY，磁碟空間的剩餘量及使用數量，各程式 PROG－RAM 的大小，〔B〕檔案的起始地址及程式的長度等等。GO還有其他的好處，當 APPLE Ⅱ 在最初啓動時，它BOOT 起 DOS 及 GO要用去10秒，當 DOS 已被BOOT起時，GO在短短的 6 秒中便可被執行，多快！另外假設磁碟內有 105 個檔案，它永不會將檔案名斷開。

程式的使用

GO 程式使用十分簡易，所有的功能都是一看便明。最好用的就是每當＜RETURN＞ 鍵被按下時，它便由第二個 LEVEL 回到第一個 LEVEL，這給予用家無限的方便。當程式被執行時，它會先做一個“CATALOG”的命令，當超過 16個 FILE 顯示在螢幕上時，程式便會稍停一下，然後在最高的一行顯示指示，如：

〔RETURN〕＝THIS SCREEN，
〔SPACE〕＝MORE FILES

按〔RETURN〕鍵表示我們想看的檔案已被顯示在螢幕上，否則按＜SPACE＞ 鍵繼續看其餘的 FILE 。假如磁碟檔案少於16個，GO 便會繼續執行它的工作。

在這時候，磁碟的標題便在螢幕頂部顯示，用家可以將行句20的T＄改變，便可成爲自己的磁碟標題，小心T＄的長度不可超過40個字，T＄會自己中心定位的。

跟着每一個檔案的最左方，會由一個英文字母所代表，而最下的一行則顯示：

FILE CHOICE？(OR #＝SPACE，X＝END)

●鍵入相關的英文字以選擇檔案（我們的程式磁碟用同方法）
●鍵入“#”去顯示磁碟空間的剩餘量及耗去量。
●鍵入“X”去完結。
●按〔RETURN〕再執行“CATALOG”的命令。

當選擇了某一個檔案後，相關的句子便會顯現：

APPLESOFT
OR INTEGER: RUN LOAD TEND SIZE
TEXT: EXEC TEND SIZE
BINARY: RUN LOAD TEND ADDRESS SIZE

假如所選的是“TEND”，另一句子又會顯現如：

LOCK UNLOCK RENAME DELETE VERIFY

以上全部是 DOS 的常用命令，當這個功能完成後，再按〔RETURN〕鍵便回到原來的選擇，再按〔RETUR－N〕便去到最早的一個選擇。順帶一提“RENAME”及“DELETE”功能會再問一次“是”或“否”（“Y”OR “N”）以確定選擇是否正確。在選擇“SIZE”的命令時，GO會顯示這個 FILE 的扇區量，如所選的是 BINARY FILE，其中之一的“ADDRESS”選擇則顯示出這個FILE的起始碼及 FILE 長度，接著GO會再BOOT 起 DOS ，再進入GO 的程式中。

GO][一個取代 HELLO 的程式

```
1  REM  **************************
2  REM  *                        *
3  REM  *          G O           *
4  REM  *  A GREETING PROGRAM    *
6  REM  *                        *
7  REM  **************************
20 T$ = "GO ][ (C) MICRO ORCHARD
   1984"
30  GOTO 280
40  REM  INPUT
50 C = H:I$ = "": FOR X = H TO H +
   L:C$(X) = "":T$(X) = "": NEXT
60  IF C = H - I THEN C = H
70  HTAB C
80  GET C$(C): IF C$(H) = R$ THEN
    HTAB I: CALL  - 868: INVERSE
   : POP : GOTO 740
90  IF C$(C) = C$ THEN C$(C) = ""
   : GOTO 70
100  IF C$(C) = F$ AND T$(C) > =
    A$ THEN 120
110  IF C$(H) < A$ OR C$(H) > Z$ THEN
    70
120  IF C$(C) = R$ THEN  CALL  -
    868: GOTO 190
```

```
130  IF C$(C) = B$ THEN T$(C - I)
      = C$(C - I):C = C - I: GOTO
     60
140  IF C = H + L + I THEN  PRINT
     BL$;: GOTO 80
150  IF C$(C) = F$ AND T$(C) = ""
      THEN T$(C) = S$
160  IF C$(C) = F$ THEN C$(C) = T
     $(C):C = C + I: GOTO 60
170  IF  ASC (C$(C)) < 32 THEN 60

180  VTAB V: HTAB C: PRINT C$(C);
     :C = C + I: GOTO 60
190  FOR X = C - I TO H STEP  - I
     : IF  ASC (C$(X)) + Z < > B
      THEN C = X: GOTO 210
200  NEXT
210  FOR X = H TO C:I$ = I$ +  CHR$
     ( ASC (C$(X)) + Z): NEXT : RETURN

220  REM  VARIABLES :: NOTE :: B3
     $ = 3 BLANKS, B40$ = 40 BLAN
     KS
230 C = 0:F0 = 40:F1 = 41:F2 = 25
     6:I = 1:P = 3:Q = 4:R = 5:B =
     160:Y = 62:H = 8:L = 29:F =
     7:Z = 128
240 A$ = "A":Z$ = "Z":B3$ = "   "
     :B40$ = "
                                  "
250  DEF  FN S(R) =  PEEK (C +  PEEK
     (F0) +  PEEK (F1) * F2): DEF
      FN T(C) =  INT ((F1 -  LEN
     (T$)) / 2 + .5)
260 D$ =  CHR$ (4):R$ =  CHR$ (13
     ):F$ =  CHR$ (21):B$ =  CHR$
     (8):S$ =  CHR$ (32):C$ =  CHR$
     (44):BL$ =  CHR$ (7): RETURN

270  REM  SETUP
280  POKE 768,32: POKE 769,12: POKE
     770,253: POKE 771,201: POKE
     772,141: POKE 773,208: POKE
     774,3: POKE 775,76
290  POKE 776,44: POKE 777,174: POKE
     778,76: POKE 779,60: POKE 78
     0,174: POKE 781,0: POKE 4460
     2,0: POKE 44603,3
300  POKE 44452,20: POKE 44605,19
     : DIM A$(22,6),C$(38),T$(38)
     : ONERR  GOTO 1060
310  GOSUB 230: PRINT D$"NOMON C,
     I,O": TEXT : HOME
320  INVERSE : PRINT B40$;: VTAB
     I: HTAB I: PRINT "<RET> = TH
     IS SCREEN, <SP> = MORE FILES
     ";: POKE 34,2
330  HOME : NORMAL : PRINT R$: VTAB
     I: PRINT D$"CATALOG":V =  PEEK
     (37) - 2: VTAB 24: PRINT  CHR$
     (10)
340  VTAB I: HTAB I: INVERSE : PRINT
     B40$;: VTAB I: HTAB  FN T(C)
     : PRINT T$
350  VTAB P: FOR X = P TO V: HTAB
     Q: FOR C = P TO R:A$(X,C) =
      CHR$ ( FN S(R)): NEXT : PRINT
     : NEXT
360  VTAB P: FOR X = P TO V: HTAB
     Q: NORMAL : PRINT B3$;: HTAB
     R: INVERSE : PRINT  CHR$ (X +
     Y): NEXT :LL = X + 61:V = X +
     I
370  REM  FILE NAME/TYPE
380  VTAB V: HTAB H: CALL  - 868:
      POKE  - 16368,0: INVERSE :N
     $ = "":E = 39
390  HTAB H: PRINT "FILE CHOICE?
     (OR #=SPACE, X=END) ";: VTAB
     V: HTAB R: GET X$: IF X$ = R
     $ THEN 310
400  IF X$ = "X" THEN  HTAB H: CALL
      - 868: PRINT "END";: POKE 4
     4602,12: POKE 44603,253: POKE
     44452,22: POKE 44605,21: GOTO
     640
410  IF X$ = "#" THEN 660
420  IF X$ < A$ OR  ASC (X$) > LL
      THEN 390
430 LV =  ASC (X$) - Y: VTAB LV: HTAB
     I:S =  PEEK (F0) +  PEEK (F1
     ) * F2
440 T =  PEEK (S + I): IF T < >
     193 AND T < > 194 AND T < >
     201 AND T < > 212 THEN  VTAB
     V: GOTO 390
450  FOR X = E TO F STEP  - I: IF
      PEEK (S + X) < > B THEN E =
     X: GOTO 470
460  NEXT
470  FOR X = F TO E:N$ = N$ +  CHR$
     ( PEEK (S + X)): NEXT : VTAB
     LV: HTAB H: PRINT N$: VTAB V
     : HTAB H: CALL  - 868
480  REM  COMMAND
490  HTAB H: IF T = 212 THEN  PRINT
     "EXEC";: GOTO 510
500  PRINT "RUN  LOAD";
510  PRINT "  TEND";: IF T = 194 THEN
      PRINT "  ADDRESS";
520  PRINT "  SIZE";
530  VTAB V: HTAB R: GET X$: IF X
```

```
$ = R$ THEN  VTAB LV: HTAB H
: NORMAL : PRINT N$: GOTO 38
0
540  IF X$ = A$ AND T = 194 THEN
      HTAB I: CALL  - 868: PRINT
     BL$;: GOTO 1140
550  IF X$ = "S" THEN 710
560  IF X$ = "T" THEN 740
570  IF X$ = "E" AND T <  > 212 OR
     X$ <  > "E" AND T = 212 THEN
     530
580  IF X$ = "E" THEN O$ = "EXEC"
     : GOTO 620
590  IF X$ = "R" THEN O$ =  RUN":
      GOTO 620
600  IF X$ <  > "L" THEN 530
610 O$ = "LOAD"
620  POKE  - 16298,0: POKE 44452,
     22: POKE 44605,21: POKE 4460
     2,12: POKE 44603,253: IF T =
     194 THEN BI$ = "B"
630  HTAB H: CALL  - 868: PRINT B
     I$O$: PRINT D$BI$O$N$;
640  NORMAL : POKE  - 16298,0: POKE
     34,0: VTAB V - (X$ = "X" OR
     T = 193 OR T = 201): END
650  REM  DISK SPACE
660  HTAB H: CALL  - 868: PRINT "
     DISK SPACE:";
670  RESTORE : FOR X = 896 TO 960
     : READ A: POKE X,A: NEXT : POKE
     905,149: POKE 938,149
680 J =  PEEK (43626) * 16:K =  PEEK
     (43624): POKE 897,J: POKE 89
     8,K: POKE 911,J: POKE 912,K:
      CALL 919
690 FR =  PEEK (917) +  PEEK (918
     ) * F2:US = 496 - FR: PRINT
      " USED="US"  FREE="FR;: HTAB
     R: GET X$: GOTO 310
700  REM  SIZE
710  HTAB H: CALL  - 868: PRINT "
     SIZE: ";: FOR X = P TO R: PRINT
     A$(LV,X);: NEXT : PRINT " SE
     CTORS";
720  HTAB R: GET X$: CALL  - 868:
      GOTO 490
730  REM  FILE TENDING
740  VTAB V: HTAB I: CALL  - 868:
      IF  PEEK (S) = 106 THEN  POKE
     S,170
750  HTAB H: PRINT "LOCK UNLOCK R
     ENAME DELETE VERIFY";: HTAB
     R: GET X$: IF X$ = R$ THEN  CALL
      - 868: HTAB H: GOTO 490
760  IF X$ = "L" THEN 840
770  IF X$ = "U" THEN 870
780  IF X$ = "R" THEN 900
790  IF X$ = "D" THEN 990
800  IF X$ <  > "V" THEN 750
810  REM  VERIFY
820  CALL  - 868: HTAB H: PRINT "
     VERIFY": PRINT D$"VERIFY"N$:
      GOTO 740
830  REM  LOCK
840  IF  PEEK (S) = 170 THEN 750
850  CALL  - 868: HTAB H: PRINT "
     LOCK": PRINT D$"LOCK"N$: POKE
     S,170: GOTO 740
860  REM  UNLOCK
870  IF  PEEK (S) = 160 THEN 750
880  CALL  - 868: HTAB H: PRINT "
     UNLOCK": PRINT D$"UNLOCK"N$:
      POKE S,160: GOTO 740
890  REM  RENAME
900  IF  PEEK (S) = 170 THEN  POKE
     S,106
910  HTAB I: CALL  - 868: PRINT "
     RENAME";: NORMAL : PRINT "="
     ;: GOSUB 50: INVERSE : IF I$
      = N$ THEN 740
920  HTAB I: PRINT BL$"OK?";: NORMAL
     : PRINT "    ";: INVERSE
930  HTAB R: GET X$: IF X$ = "N" OR
     X$ = R$ THEN 740
940  IF X$ <  > "Y" THEN 930
950  HTAB I: PRINT "RENAME";: NORMAL
     : PRINT "=": IF  PEEK (S) =
     106 THEN UN = I: PRINT D$"UN
     LOCK"N$
960  PRINT D$"RENAME"N$C$I$: IF U
     N THEN UN = 0: PRINT D$"LOCK
     "I$
970 N$ = I$: VTAB LV: HTAB H: CALL
      - 868: INVERSE : PRINT N$: GOTO
     740
980  REM  DELETE
990  HTAB I: CALL  - 868: PRINT "
     OK?";: NORMAL : PRINT "  ";
     : INVERSE : PRINT BL$"DELETE
     ";
1000  IF  PEEK (S) = 170 THEN  POKE
     S,106: PRINT "LOCKED FILE"BL
     $;
1010  HTAB R: GET X$: IF X$ = "N"
     OR X$ = R$ THEN 740
1020  IF X$ <  > "Y" THEN 1010
1030  HTAB I: NORMAL : PRINT "
     ": IF  PEEK (S) = 106 THEN  PRINT
     D$"UNLOCK"N$
1040  PRINT D$"DELETE"N$: GOTO 31
     0
```

GO II 程式

```
1050  REM  ERRORS
1060  POKE 768,104: POKE 769,168:
      POKE 770,104: POKE 771,166:
      POKE 772,223: POKE 773,154:
      POKE 774,72
1070  POKE 775,152: POKE 776,72: POKE
     777,96: CALL 768: POKE 216,0
     : VTAB V: HTAB I: CALL  - 86
     8
1080 ER =  PEEK (222): VTAB V: HTAB
     H: INVERSE : PRINT BL$"ERROR
      "ER;: IF ER = 1 THEN  PRINT
     " - LANGUAGE NOT AVAILABLE";
1090  IF ER = 6 THEN  PRINT " - F
     ILE NOT FOUND";
1100  IF ER = 8 AND X$ = "V" THEN
      PRINT " - UNABLE TO VERIFY"
     ;: GOTO 1120
1110  IF ER = 8 THEN  PRINT " - I
     /O";
1120  HTAB R: GET X$: RUN
1130  REM  ADDRESS/LENGTH
1140  HTAB I: PRINT "OK?";: NORMAL
     : PRINT "      ";: INVERSE : PRINT
     "ADDRESS/LENGTH  (BOOT IF FA
     ILS)";: HTAB R: GET X$
1150  IF X$ = "N" OR X$ = R$ THEN
      HTAB 1: CALL  - 868: GOTO 4
     90
1160  IF X$ <  > "Y" THEN 1140
1170  ONERR  GOTO 1220
1180  HTAB 1: NORMAL : PRINT "
     ": INVERSE : PRINT D$"BLOAD"
     N$
1190  LOMEM:  PEEK (115) +  PEEK
     (116) * 256 - 2000:H = 5216 +
      PEEK (115) +  PEEK (116) *
     256
1200 A =  PEEK (43634) +  PEEK (4
     3635) * 256:L =  PEEK (H) +
      PEEK (H + 1) * 256
1210  VTAB  PEEK (37): HTAB 8: CALL
      - 868: PRINT "BLOAD ADDRESS
     ="A"   LENGTH="L;
1220  HTAB 5: GET X$: CALL  - 868
     : HTAB 8: PRINT "REQUIRED BO
     OT - PLEASE WAIT"
1230  NORMAL : VTAB  PEEK (37) -
     1: PRINT  CHR$ (4)"PR#6"
1240  DATA 1,96,1,0,17,0,145,3,0,
     149,0,0,1,0,0,96,1,0,1,239,2
     16,0,0,169,3,160,128,32,217,
     3,162,0,142,149,3,142
1250  DATA 150,3,160,56,185,0,149
     ,162,8,10,144,8,238,149,3,20
     8,3,238,150,3,202,208,242,20
     0,192,196,144,232,96
1260  REM  BY MICRO ORCHID
```

RB5X 的程式語言，TINY BASIC，有一些優良的特點，一個小型套裝內就包含了強大的功能。當設計 RB5X 時，已考慮到要創造一種自然語言，去控制機械人，才會爲社會公衆人士所接受。所以令人最關注的問題是：“何時才能將這個構思實現呢？”

機械人控制語言

第一步已開展得甚爲迅速，在一九八三年六月，RB 機械人製作人員研究由 EXCALIBUR TECHNOLOGIE—S CORPORATION 所提供的軟件系統，稱爲 SAV—VY 。該系統顯示淺白語言程式與電腦溝通有重大成果。當 SAVVY 首次公諸於世時，電腦溝通起了突破，證明可改由從代碼轉用淺白語言。

梗概而言，SAVVY 系統結合電腦操作系統，使電腦無窮無盡地擴大了它的用途與應用領域。

SAVVY 結合補充性模式認知過程（簡稱 APRP）可令它對不準確的輸入，程式名稱或指示，作出反應。舉例：如果 SAVVY 已認知 BLOW THE HORN（响按）這個指示，它可以不用再需教導，便會明白串錯字或句子錯誤的指示，如 BLWO YOUR HRON，BLWO THEH OR—N 等等。

在 SAVVY 系統，任何指定的輸入可以連繫上任何指定的輸出。所以，可以訓練 SAVVY系統，當任何的指示，類似 BLOW THE HORN ，就會輸出該指示。再者，SAVVY可以學習明白 HONK，SOUND WARNING SIGNALS（發出警告訊號）及 OBLAY DER ORN—HAY 的意義均與 BLOW THE HORN 相同。如果你出正確的代碼。

RB ROBOT 公司已與 EXCALIBUR 簽訂合約，研究發展新的機械人控制語言（ROBOT CONTROL LANGUAGE 簡稱 RCL）使用 SAVVY 系統及蘋果Ⅱ號電腦。所有程式是用淺白語言所寫，而自動編譯成爲 TI—NY BASIC 代碼，輸入機械人電腦內。

用簡單語言控制

用機械人控制語言（RCL）加上 SAVVY 系統，RB5X 的程式編寫員不用牢記向前移動的指令，究竟是@#7802=#9 抑或是@ 7801=#20；他只須打上 MOVE FORWARD RCL 用簡單英語替代了電腦代碼。程式編寫員可以隨意在任何時間增加 RCL 的詞匯。事實上，程式編寫員可用該系統創造新語言，並且增加現存指令的同義詞，及創造新指令更擴展語言的應用。

如果操作者，用RCL 寫常式令機械人旋轉移動，這項常式（例如：RUN IN CIRCLES 或 MAKE A CIRCLE）便漸漸變成 RCL 詞匯的一部份。RUN IN CIRCLES 逐步成爲一如 SAVVY 基本語言指令的功能，好像 ADD，EDIT或SAVE 等。例如，操作者想機械人一面哼着調子，一面跳舞，可以指示機械人 MOVE FO—RWARD，RUN IN CIRCLES，MOVE BACK—WARD。你無須再提醒電腦找尋 RUN IN CIRCLES，SAVVY 會自動作出指示無需用要求其他詳細指示。

自動編寫程序員

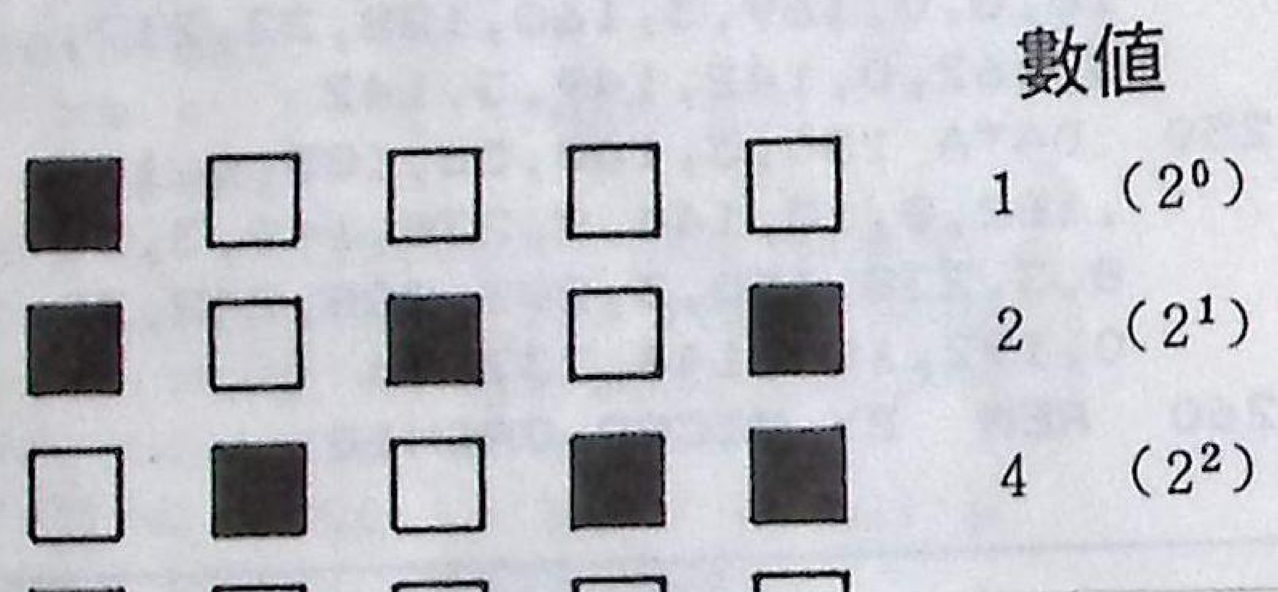
你可想像SAVVY系統是自動編寫程序員，它清楚電腦

布倫時鐘

傳統以來，時鐘報時的方法，都是以時針加分針及秒針來顯示當時的時間，在七十年代，電子工業之發達，時分秒又以數字管所取代，在八十年代的今天，電子電腦科技的進步，我們試圖把這個傳統掉在一旁而改用另一個方法去顯示時間，這就是我們今次的主題——二進制布倫時鐘。

■程式

本程式是用來驅動布倫時鐘的，它將會在 APPLE Ⅱ 的低解像畫面作出顯示。實際所顯示的時間是經過幾個 LOOP 來作 TIME DELAY，從而可以精確地計算出一秒與一秒間的時間，至於這幾個 DELAY LOOP 的計算方法及調校，會在下面論述。當程式在執行時，低解像度畫面便會出現一幅 4 × 5 的圖案，如下圖所示：



制的關係，數値的計算，由上而下爲1，2，4，8，而實
際的數値爲有顏色的方塊的總和，如圖所示，時間就是11時
42分46秒。

■程式的使用

當程式被執行時，畫面首先會被消除，然後就是時間的
調定，我們要逐一按入時分秒的數值，再按〔＜RETU—
RN＞〕鍵，畫面會再度消除，跟着程式會再等第二次按鍵
便開始執行，這個做法，是可以令到用家更準確地調校實際
的時間。程式會不斷的運行，直至（CTRL－RESET）
被按下爲止。

■時間延遲的計算

這個程式最初是以 APPLESOFT BASIC 編寫的，但發覺它未能與眞正的時間配合，若非太快，便是太慢。因爲 BASIC 要經由 INTERPRETER 解譯才執行，所以很難作時間上的配合。後來改用 ASSEMBLY 語言，效果便好多了。在這程式中的調校，筆者耗去了大半天的時間才 SET 好，但這並不表示每一個 APPLE Ⅱ，也能準確地執行這個計時的方法，需要讀者去稍作調校才行。數值的大小可從程式中的地址＄0803至＄0806 找出，第一個數值（＄47）是粗略的計算值，第二個數値（＄48）是 DEL

```
O  NORMAL : POKE  - 16298,0: POKE
   34,0: VTAB V - (X$ = "X" OR
   T = 193 OR T = 201): END
O  REM  DISK SPACE
O  HTAB H: CALL  - 868: PRINT "
   DISK SPACE:";
)  RESTORE : FOR X = 896 TO 960
   : READ A: POKE X,A: NEXT : POKE
   905,149: POKE 938,149
)  J =  PEEK (43626) * 16:K =  PEEK
   (43624): POKE 897,J: POKE 89
   8,K: POKE 911,J: POKE 912,K:
   CALL 919
)  FR =  PEEK (917) +  PEEK (918
   ) * F2:US = 496 - FR: PRINT
   " USED="US"  FREE="FR;: HTAB
   R: GET X$: GOTO 310
   REM  SIZE
   HTAB H: CALL  - 868: PRINT "
   SIZE: ";: FOR X = P TO R: PRINT
   A$(LV,X);: NEXT : PRINT " SE
   CTORS";
   HTAB R: GET X$: CALL  - 868:
   GOTO 490
   REM  FILE TENDING
   VTAB V: HTAB I: CALL  - 868:
   IF  PEEK (S) = 106 THEN  POKE
   S,170
   HTAB H: PRINT "LOCK UNLOCK R
   ENAME DELETE VERIFY";: HTAB
   R: GET X$: IF X$ = R$ THEN  CALL
   - 868: HTAB H: GOTO 490
   IF X$ = "L" THEN 840

     RENAME";: NORMAL : PRINT "="
     ;: GOSUB 50: INVERSE : IF I$
      = N$ THEN 740
920  HTAB I: PRINT BL$"OK?";: NORMAL
     : PRINT "     ";: INVERSE
930  HTAB R: GET X$: IF X$ = "N" OR
     X$ = R$ THEN 740
940  IF X$ <  > "Y" THEN 930
950  HTAB I: PRINT "RENAME";: NORMAL
     : PRINT "=": IF  PEEK (S) =
     106 THEN UN = I: PRINT D$"UN
     LOCK"N$
960  PRINT D$"RENAME"N$C$I$: IF U
     N THEN UN = 0: PRINT D$"LOCK
     "I$
970  N$ = I$: VTAB LV: HTAB H: CALL
      - 868: INVERSE : PRINT N$: GOTO
     740
980  REM  DELETE
990  HTAB I: CALL  - 868: PRINT "
     OK?";: NORMAL : PRINT "  ";
     : INVERSE : PRINT BL$"DELETE
     ";
1000  IF  PEEK (S) = 170 THEN  POKE
     S,106: PRINT "LOCKED FILE"BL
     $;
1010  HTAB R: GET X$: IF X$ = "N"
     OR X$ = R$ THEN 740
1020  IF X$ <  > "Y" THEN 1010
1030  HTAB I: NORMAL : PRINT "
     ": IF  PEEK (S) = 106 THEN  PRINT
     D$"UNLOCK"N$
1040  PRINT D$"DELETE"N$: GOTO 31
     0
```

機械人控制語言

機械人特輯



RB5X 私人機械人

RB5X 身裁矮小，僅兩尺半高。身軀圓筒型，頭部是圓拱型（見照片1）。它有兩個輪及平衡翼，可以三百六十度轉動，前後移動的速度是每秒鐘四吋。環身裝有多個按鈕，組成聲波導向系統，可以感應到十吋至三十五呎內的物體及屏障。體內的紅外綫感應器可探測地下的記號，帶引它去補充電源（見照片2）。

RB5X 是私人機械人，是 RB ROBOT CORPO—RATION 系列第一個產品。亦是市場上「投石問路」探測反應的先驅者。

機械人可裝置上吸塵機拖架，導航羅盤，火警探測器及滅火器等。如果加設語言認知儀器，RB5X 更可以聽從召喚及對口語命令作出有效反應。

機械人初期設計是使用(NATIONAL SEMICOND—UCTOR 的 INS—8073 8—BIT 處理器，及 TI—NY BASIC 在 ROM內）及 8000BYTE RAM，控制機械人。記憶量可以擴充至 16000 BYTE RAM 及 4K—BYTE EPROM。

機械人預料將與人類溝通更爲便利及友善，比之在市場上大量推出的電腦更勝一籌。

RB5X 的程式語言，TINY BASIC，有一些優良的特點，一個小型套裝內就包含了強大的功能。當設計 RB 5X 時，已考慮到要創造一種自然語言，去控制機械人，才會爲社會公衆人士所接受。所以令人最關注的問題是："何時才能將這個構思實現呢？"

機械人控制語言

第一步已開展得甚爲迅速，在一九八三年六月，RB 機械人製作人員研究由 EXCALIBUR TECHNOLOGIE—S CORPORATION 所提供的軟件系統，稱爲 SAV—VY。該系統顯示淺白語言程式與電腦溝通有重大成果。當 SAVVY 首次公諸於世時，電腦溝通起了突破，證明可改由從代碼轉用淺白語言。

梗概而言，SAVVY 系統結合電腦操作系統，使電腦無窮無盡地擴大了它的用途與應用領域。

SAVVY 結合補充性模式認知過程（簡稱 APRP）可令它對不準確的輸入，程式名稱或指示，作出反應。舉例：如果 SAVVY 已認知 BLOW THE HORN（响按）這個指示，它可以不用再需教導，便會明白串錯字或句子錯誤的指示，如 BLWO YOUR HRON，BLWO THEH OR—N 等等。

在 SAVVY 系統，任何指定的輸入可以連繫上任何指定的輸出。所以，可以訓練 SAVVY系統，當任何的指示，類似 BLOW THE HORN，就會輸出該指示。再者，SAVVY可以學習明白 HONK，SOUND WARNING SIGNALS（發出警告訊號）及 OBLAY DER ORN—HAY 的意義均與 BLOW THE HORN 相同。如果你打出 SOUND THE HORN，TINY BASIC 亦會輸出正確的代碼。

RB ROBOT 公司已與 EXCALIBUR 簽訂合約，研究發展新的機械人控制語言（ROBOT CONTROL LANGUAGE 簡稱 RCL）使用 SAVVY 系統及蘋果Ⅱ號電腦。所有程式是用淺白語言所寫，而自動編譯成爲 TI—NY BASIC 代碼，輸入機械人電腦內。

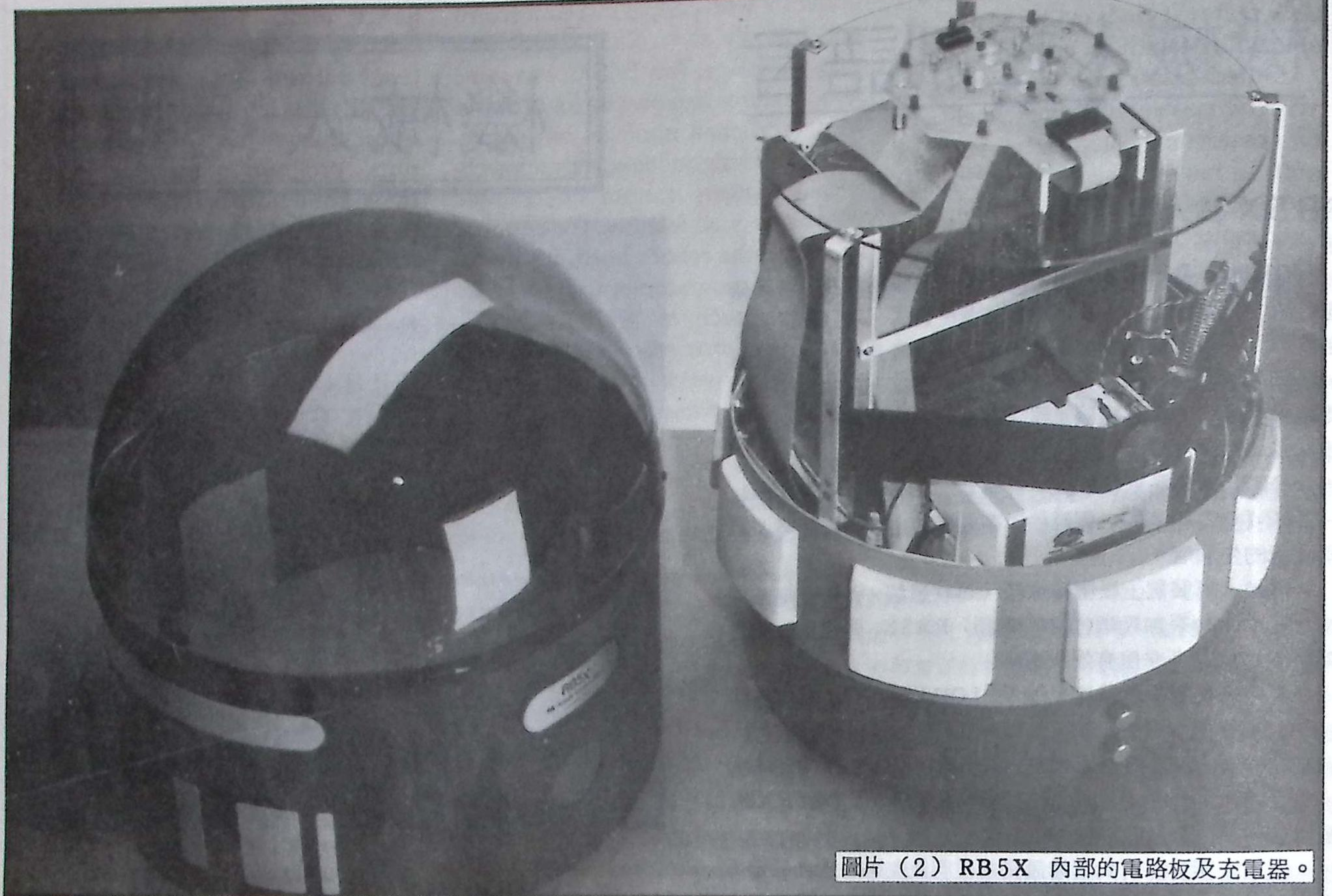
用簡單語言控制

用機械人控制語言（RCL）加上 SAVVY 系統，RB5X 的程式編寫員不用牢記向前移動的指令，究竟是@#7802=#9 抑或是@ 7801=#20；他只須打上 MOVE FORWARD RCL 用簡單英語替代了電腦代碼。程式編寫員可以隨意在任何時間增加 RCL 的詞匯。事實上，程式編寫員可用該系統創造新語言，並且增加現存指令的同義詞，及創造新指令更擴展語言的應用。

如果操作者，用RCL 寫常式令機械人旋轉移動，這項常式（例如：RUN IN CIRCLES 或 MAKE A CIRCLE）便續漸變成 RCL 詞匯的一部份。RUN IN CIRCLES 逐步成爲一如 SAVVY 基本語言指令的功能，好像 ADD，EDIT或SAVE 等。例如，操作者想機械人一面哼着調子，一面跳舞，可以指示機械人 MOVE FO—RWARD，RUN IN CIRCLES，MOVE BACK—WARD。你無須再提醒電腦找尋 RUN IN CIRCLES，SAVVY 會自動作出指示無需用要求其他詳細指示。

自動編寫程序員

你可想像SAVVY系統是自動編寫程序員，它清楚電腦



圖片（2） RB5X 內部的電路板及充電器。

系統內所有的細節，它提示電腦每一項工作的次序，資料種類，連繫等等。SAVVY告訴電腦何時打開或關上檔案，及如何將資料從系統一點移到另一點。 SAVVY 比之人類更熟知蘋果電腦或 IBM PC系統內部運作情況。它跟隨編寫員的指示，一步一步將程式轉爲代碼。

SAVVY與程式編寫員必需能夠溝通。增加RCL 的詞匯，可令兩者溝通益見完善。編寫員可以用自己認爲較佳的字眼去進行指示。

一些技巧純熟的編寫員，有充份能力處理意義隱晦的代碼，可選擇簡短的字母作爲 RCL 的詞匯。例如教導系統 MF 意思相同於 MOVE FORWARD，MB 等於 MOVE BACKWARD ，RIC 即是 RUN IN CIRCLES 的意思。利用此種技巧，無須隱藏代碼的意義，因爲 SAVVY 運用其原有的詞匯，而演譯每一項指令。

SAVVY 的詞匯不斷擴大，因爲新的基本指示不斷的加入，以運用機械的手臂，感應系統及其它周邊。

RCL 程式編寫員可以加入指示，更新系統，只須簡單地重新給每一個字眼新定義。 RCL 用家可以自行發展自己一套的新基本指示，加入個人的系統內。到目前爲止，RCL 用家仍未有共通「字典」。彼此根據自己的需要，訂定控制機械人的指示。

SAVVY系統的動力主要視乎 TINY BASIC 軟件硬件的本質。 NATIONAL SEMICONDUCTOR 的處理器，記憶量及支綏綫路軟件包成爲 TINY BASIC 引擎。

RCL 發展的目標是創製淺白語言編寫程式的環境使用家易於控制機械人的習慣。他們不想限制ＲＢ５Ｘ硬件的實用特點，而要盡量善用TINY BASIC。他們目的是創製有彈性，易於擴展的編寫程式架構。

INS－8073 直接支持 RS－232C去與 RB5X 內的 TINY BASIC 解譯／監督程式溝通。很明顯，一個正確運作的機械人，是無需要用兩種途徑去與 TINY BASIC 溝通，我們只要一項途徑，就是直接將程式由 APPLE 電腦，轉往機械人。

語言結構

溝通途徑解決的另一項關鍵是 TINY BASIC 程式儲存。典型的 BASIC 句子的句法是句子的行號，然後是句子的內容。所以一個檔案夾稱爲 SOURCE，它有一個類目（ ITEM）是STATEMENT NUMBER。另外兩個類目是 BASIC TEXT及 MEANING。BASIC TEXT 是程式句子，而 MEANING 是判斷分析及教育性記號。

SAVVY 的結構與一般不同，用〝FOLDER〞替代〝FILE〞，PAGE 取代〝RECORD〞，ITEM代〝FIELD〞及 TASK 代〝PROGRAM〞。在 RCL，儲存於 TINY BASIC句子的檔案結構（ FOLDER STRUCTURE）是

```
SOURCE (a folder)
1 Id   STATEMENT NUMBER
2 Item BASIC TEXT
3 Item MEANING
```

用家能夠以英語跟電腦直接溝通 極之口語化結構的SAVVY

SAVVY工作（TASK)是由 SAVVY工作編輯所決定，RCL "基本"的編寫程式元素是採用上文已提過的指令例如：

```
GO FORWARD
SPIN RIGHT 90 DEGREES
WAIT this many seconds<1>
TURN ON HORN
SAY this phoneme<1>
```

從這些分割的指令，SAVVY 工作編輯配合 TINY BASIC句子就會進行指定的功能。

以下是編輯 TINY BASIC 的過程。

```
INCREMENT STATEMENT NUMBER(a task)
1 Dose ADD the STATEMENT NUMBER and
  STATEMENT NUMBER INCREMENT VALUE
2 and COPY from SUM to STATEMENT
  NUMBER
3 and END

COMPILE the BASIC statement
        <1> which means <2>
      (a function)

1 Does INCRREMENT STATEMENT NUMBER
2 and COPY from (1) to BASIC TEXT
3 and COPY from (2) to MEANING
4 and SAVE new page in SOURCE
5 and END
```

一些RCL 基本元素非常簡單，例如 STOP ALL MOTION (工作) 即是 COMPILE BASIC STA—TEMENT @#7802=#0（意思是停止所有動作）。在另一個例子，元素 WAIT 可作邏輯上決定某一特定停頓時刻是否超一秒鐘（在很多情況下，RB5X 是需要等待一段時間才進行另一動作），如果等待時間是少於一秒，RCL 就會發出" DELAY nnn " nnn代表百萬分之一秒。

這是應用RCL 元素的一般模式，而較高層次的 RCL 元素的例子是：

```
HONK the horn for
this many seconds <1>

1 Does TURN ON HORN
2 and WAIT this many
  seconds<1>
3 and TURN OFF HORN
4 and END
```

這裡，不再再詳盡解譯 SAVVY ，只希望略爲涉覽在 RCL 用 SAVVY 編譯 TINY BASIC 的過程。再重申一次，一般 RCL 用家是絕少機會看到 TINY BASIC 代碼。事實上，亦不會看到淺白語言 SAVVY 工作。RCL 的重點是使機械人操作者用簡單的方法告訴 RB5X 做些甚麽。

表1所舉 RCL 常式（ROUTINE）及表二是列舉 TINY BASIC常式足可提供示範比較。毫無疑問 RCL 加上 SAVVY 的研究創新突破，而 RB ROBOT 機構已陸續用 RCL 發展應用程式，以擴大 RB5X 的效能。

摘要

RCL及 SAVVY 比之 TINY BASIC代碼有更大的改進。程式編寫員（及機械人）仍要倚賴外在電腦創造程式，然後加上機械人內。明顯地，RB5X 仍需要很多腦袋，故此現在已研究製造一張咭令 RCL 可在 RB5X 內編寫程式。同時希望 SAVVY 系統能學習感應輸入模式而作出適當的反應，隨着經驗的累積，相信RCL 會愈來愈完整，而系統可在多方面擴展。

RCL 另一項特點是可用相同淺白語言指示與任何語言去編輯程式（包括 FORTRAN，滙編語言）RCL 詞匯不斷累積豐富後，將可控制鍊鋼器及機械廠的工作。RCL 配合SAVVY初試啼聲已表示威力巨大，它勢將成爲一般用途機械人的語言。□

```
1 RESET THE ROBOT
2 INITIALIZE MEMORY
3 BEGIN A REPEATING PROCEDURE
4   INITIALIZE MESSAGES
5   CLEAR ALL VARIABLES
6   BEGIN A REPEATING PROCEDURE
7     GO FORWARD
8     LEAVE IF ANY BUMPER TOUCHED
9     DOES THE variable R compare = to ZERO
10      LEAVE THIS REPEATING PROCEDURE
11      END TEST
12    REPEAT THIS REPEATING PROCEDURE
13  LEAVE IF TAPE IS SENSED
14  MOVE WITH BETA INTELLIGENCE
15  REPEAT THIS REPEATING PROCEDURE
16 FOLLOW TAPE
17 MAINTAIN CHARGE
```

列表（1）用 SAVVY 編寫的機械人控制語言例子。程式指示機械人自行依據藏在地板的金屬條尋找充電源。

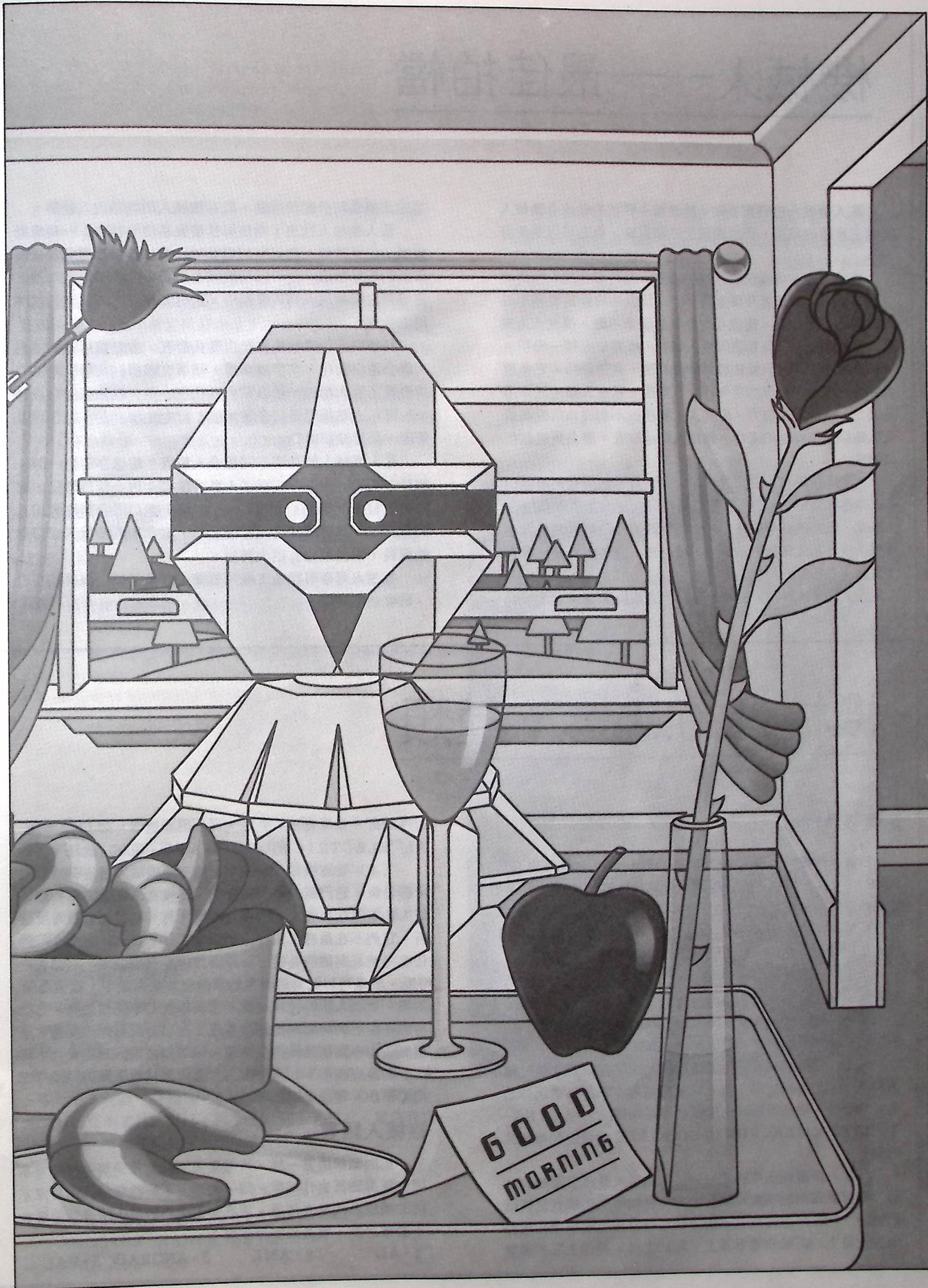
RB5X 機械人製造商
RB ROBOT CORPORATION,
SUITE 310, 18301 WEST 10TH AVE.
GOLDEN, CO 80401, (303) 279—5525.

Tiny BASIC Text	Meaning of Text
10 T=10	Number of whole seconds
20 GOSUB 1000	Go wait
30 T=10	Number of whole seconds
40 GOSUB 1000	Go wait
50 @#7803=#98	Initialize I/O
60 N=TOP	Initialize experience block
70 O=TOP+#FF	Initialize inhibition block
80 M=TOP+#200	
90 FOR P=N TO M	
100 @P=#FF	
110 NEXT P	
120 REM START A LOOP	Begin A Loop
130 @#7803=#98	Initialize I/O
140 CLEAR	Clear variables
150 REM START A LOOP	Begin A Loop
160 @#7802=#09	Go forward
170 Y=@#7800	Test for bumper contact
180 IF Y<255 GOTO 230	Exit if any contact
190 IF R<0 GOTO 210	Make a comparison
200 GOTO 230	Exit this Loop
210 REM END HERE	Fall through to here
220 GOTO 150	Repeat this Loop
230 REM EXIT TO HERE	
240 X=#02	Turn on LED 1
250 GOSUB 1100	Go turn on a bit
260 R=@#7802 AND #40	Test for tape sense
270 IF R=0 GOTO 310	Exit if tape sensed
280 GOSUB 2000	Go to Beta Subroutine
290 REM RETURN GOES HERE	Return to here
300 GOTO 120	Repeat this Loop
310 REM EXIT TO HERE	
320 REM START A LOOP	Begin A Loop
330 @#7803=#98	Intialize I/O
340 CLEAR	Clear variables
350 REM START A LOOP	Begin A Loop
360 @#7802=#08	Right forward
370 X=#02	Turn on LED 1
380 GOSUB 1100	Go turn on a bit
390 R=@#7802 AND #40	Test for tape sense
400 IF R=0 GOTO 440	Exit if tape sensed
410 Q=@#7802 AND #20	Test for charger contact
420 IF Q=0 GOTO 440	Exit if charger sensed
430 GOTO 350	Repeat this Loop
440 REM EXIT TO HERE	
450 Q=@#7802 AND #20	Test for charger contact
460 IF Q=0 GOTO 800	Exit if charger sensed
470 X=#40	Turn on flashing lights
480 GOSUB 1100	Go turn on a bit
490 T=0	Initialize
500 DO	Do the following
510 T=T+1	Math function
520 @#7802=#09	Go forward
530 R=@#7802 AND #40	Test for tape sense
540 UNTIL (R<>0) OR (T>=100)	Check exit conditions
550 DELAY 100	Short wait
560 @#7803=#98	Initialize I/O
570 CLEAR	Clear variables
580 REM START A LOOP	Begin A Loop
590 @#7802=#01	Left forward
600 X=#02	Turn on LED 1
610 GOSUB 1100	Go turn on a bit
620 R=@#7802 AND #40	Test for tape sense
630 IF R=0 GOTO 670	Exit if tape sensed
640 Q=@#7802 AND #20	Test for charger contact
650 IF Q=0 GOTO 670	Exit if charger sensed
660 GOTO 580	Repeat this Loop
670 REM EXIT TO HERE	
680 Q=@#7802 AND #20	Test for charger contact
690 IF Q=0 GOTO 800	Exit if charger sensed
700 X=#40	Turn on flashing lights
710 GOSUB 1100	Go turn on a bit
720 T=0	Initialize
730 DO	Do the following

Tiny BASIC Text	Meaning of Text
740 T=T+1	Math function
750 @#7802=#09	Go forward
760 R=@#7802 AND #40	Test for tape sense
770 UNTIL (R<>0) OR (T>=100)	Check exit conditions
780 DELAY 100	Short wait
790 GOTO 320	Repeat this Loop
800 REM EXIT TO HERE	
810 GOSUB 2310	Charge maintain routine
1000 FOR S=1 TO T	Wait T seconds subroutine
1005 DELAY 1000	
1010 NEXT S	
1015 RETURN	
1100 U=@#7801	Turn on a bit @#7801
1105 U=U OR X	
1110 @#7801=U	
1115 RETURN	
1120 U=@#7801	Turn off a bit @#7801
1125 U=U AND X	
1130 @#7801=U	
1135 RETURN	
2000 P=@#7800	Bumper pressed
2010 IF P=255 THEN P=251	Sonar treated as bumper #1
2020 V=@(N+P)	Check experience block
2030 IF V<>#FF GOTO 2100	Try action
2040 V=RND(1,14)	Pick random action
2050 IF (V=3) OR (V=12) OR (V=(0+P))	Pick another
2100 @#7802=V	Try action
2110 T=2	Number of whole seconds
2120 GOSUB 1000	Go wait
2130 D=200	Cancel sonar test
2140 IF (@#7800<>#FF) OR (D<95) GOT	Not successful
2150 @(N+P)=V	Successful
2160 CLEAR	Clear variables
2170 GOTO 290	Back to main program
2200 @(0+P)=V	Not successful
2210 GOTO 2040	
2310 REM START A LOOP	Begin A Loop
2320 REM START A LOOP	Begin A Loop
2330 Q=@#7802 AND #20	Test for charger contact
2340 IF Q=0 GOTO 2560	Exit if charger sensed
2350 P=0	Initialize
2360 REM START A LOOP	Begin A Loop
2370 P=P+1	Increment Loop count
2380 @#7802=#09	Go forward
2390 DELAY 100	Short wait
2400 @#7802=0	Stop all motion
2410 Q=@#7802 AND #20	Test for charger contact
2420 IF Q=0 GOTO 2540	Exit if charger sensed
2430 IF P<>5 GOTO 2520	Make a comparison
2440 @#7802=#06	Reverse
2450 DELAY 1000	Short wait
2460 @#7802=#09	Go forward
2470 T=1	Number of whole seconds
2480 GOSUB 1000	Go wait.
2490 DELAY 100	Milliseconds to wait
2500 @#7802=0	Stop all motion
2510 P=0	Initialize
2520 REM END HERE	Fall through to here
2530 GOTO 2360	Repeat this Loop
2540 REM EXIT TO HERE	
2550 GOTO 2320	Repeat this Loop
2560 REM EXIT TO HERE	
2570 X=#04	Turn on LED 2
2580 GOSUB 1100	Go turn on a bit
2590 DELAY 1000	Short wait
2600 @#7803=#98	Initialize I/O
2610 GOTO 2310	Repeat this Loop
2620 REM EXIT TO HERE	

END OF PROGRAM

列表（2）編譯成爲 TINY BASIC 代碼的同一程式。程式是由 APPLE II 轉移過 RB5X 內的電腦


GOOD
MORNING

機械人——最佳拍檔

私人機械人已研究多年，數據電子科技的發達令機械人可裝置複雜的感應，資訊處理及控制系統，令它可以運動自如。

機械人漸有成爲家庭寵物的趨勢。它可以是成人的良伴，表演藝人，溝通者及保護者。硬件及軟件的長足發展令機械人變得多才多藝。寵物在現今社會價值不靡。機械人未來將一如各種犬隻，波斯貓等受人眷顧，成爲家庭裡一份子。

當機械人有一天能夠替人解決問題，毫無疑問，它會獲得人類的歡迎。就等如現今的私人電腦，它在人類生活中用途廣泛，提供不少助力，商界人士實在找不到理由拒絕購買私人電腦去減輕他的工作。機械人長遠來看，潛力更超越私人電腦。

"私人"機械人的定義如何，可以將電腦與私人電腦的區別作爲解釋例子。在私人電腦出現之前，大型電腦只能由大企業，政府機構等擁有，大型電腦的工作只純爲大機構而服務。至於私人電腦，就等如一枝鉛筆，袋裝計算機，甚至是電話，是私有的及個人的。

而大企業的機械人主要是用於生產系統方面，它的工作是由大機構所分配及控制。私有機械人則純爲個人效勞。

私人機械人代表了兩條科技發展路綫的結合：一是自動機械，二是電腦。你可以認爲機械人是加上自動導向及控制系統的活動機器，你可以稱機械人是可以行走移動的電腦。甚至可以稱機械人爲人造動物，對外間世界作爲反應而能夠獨立生存。

科學家近年來積極指向自動化發展，並已贏得重大成果。辦公室自動化，文字處理器，通訊文儀器材及私人電腦等等發揮了巨大功能。辦公室自動化在小型企業產生極佳成績，一部私人電腦可担任多種功能：訂定生產計劃，通訊地址管理，品質控制等等。

私人機械人的功能亦同樣令人驚訝。假想你經營一間小型裝嵌工場，十至十五個工人負責將零件併合成爲產品，繼而包裝付運予顧客。營運上並不複雜，但仍需提高注意工人的責任感，品質控制及生產綫暢順，當你加入機械人成爲你的僱員，情況就會有很大轉變。

機械人可學習認識工廠的結構，計劃及其他僱員的身份，社會成份等等。

機械人的定義及種類

定義及特性

根據美國機械人研究協會(ROBOT INSTITUTE OF AMERICA）機械人的定義爲一個可以被編排程序，多用途的機器，而其目的在於幫助人類在不同的工作環境中搬運機器、零件、工具或執行一些危險的工作如炸藥、化學試驗等。機械人大致又可分爲四類：

（1）木頭機械人——這類機械人是最簡單的，它們沒有觸覺，只能做些簡單的動作，如用機械臂鉗起物件，經由微電腦發出的指令，作前、後、左、右的動作等等，HEB—OT II及 U—235等，可以說是這類機械人的例子。

（2）程序控制式——這類機械人除了有觸覺之外，還有嗅覺，可以辨別方向，察覺周圍環境的空氣污染情況作出適當的警告，更能對物件的形狀有概略的掃描，如 CINC—INNATI MILACRON 的 CONCITE 就是這類機械人的例子。

（3）多重程序控制式——這類機械人是第二類的改良品，它有更廣濶的嗅覺及觸覺能力，對物件的形狀有更精細的描繪，物件的方圓及其不同形狀之組合皆可精細地描繪於機械人身上的高解像顯示器上。除此之外，再加上它的多重程序設計，做事更多、更好，而且絕無遺漏。如 GENER—AL ELECTRIC 的 PRAGMA A—3000 就是。

（4）智慧機械人——以上的三類機械人皆沒有智慧的，即是說，它們雖有辨別的能力，却沒有決定能力。智慧機械人除了包括以上三類機械人的特性外，它還可以思考及分析。它們不在是在工廠裡做低微的工作，反之，它們執行的任務可能是與國防有關，如深海佈雷、反雷達等。它們有一特點，就是可以把曾經發生的事情記於記憶體中，經驗連續累積，就似人腦的特性一樣。當有相同的事情發生時，它們可以在極短的時間內把經驗重溫，加以分析再作出反應，所以其命中率應該是極高而準確。而這類機械人可能要九十年代才能發展成形。如現時五角大廈尙在試驗階段的 R2 D2 或C3 PO 等就是這類機械人的代表。

機械人語言

正如電腦語言一樣，人類要與不同的電腦溝通就要用不同的電腦語言去作橋樑，同一道理，不同的機械人就要用不同的機械人語言去溝通，現在發展中的機械人語言就已經有十數種之多，順英文字母序爲：

1、AL　2、AML　3、ANORAD　4、VAL

當其他工人將零件跌到地上，機械人可以幫手檢拾起來，減少了浪費。

機械人可變成你的忠誠僕人，對你的訓令一一遵從，機械人隨時跟在你身旁，為你拿東西，數東西及紀錄資訊，指出存貨狀況，那些貨品生產數量未符訂單要求，那些貨品已超額。機械人並可牢記所有複雜及瑣碎的事項，隨時提醒他的主人。而最重要是它可以徹夜不眠，不停工作，任何時刻聽從召喚效勞。

機械人甚至可聽電話，甚至作出選擇（例如你向機械人訓令如陳先生打電話來，立即駁給我，它就會照做），機械人亦可替你發信，將鍵盤及文字處理器顯示器整理好，供你使用。

機械人可以成為極佳的資料庫經理，與你溝通。它的聰穎頭腦隨時學習，只要做過一次，下次同樣工作，不用訓令，就為你做得妥妥當當。經驗不斷累積，更形可貴。機械人變成最佳拍檔。

到目前為止，已面世的機械人的功能仍只是有限度，但不斷的研究及創新突破，將令他有長足發展。

5、EMILY　6、FUNKY　7、HELP　8、MAPLE
9、MBL　10、PAL　11、RCL　12、RPL
13、SIGLA　14、T3　15、AUTOPASS

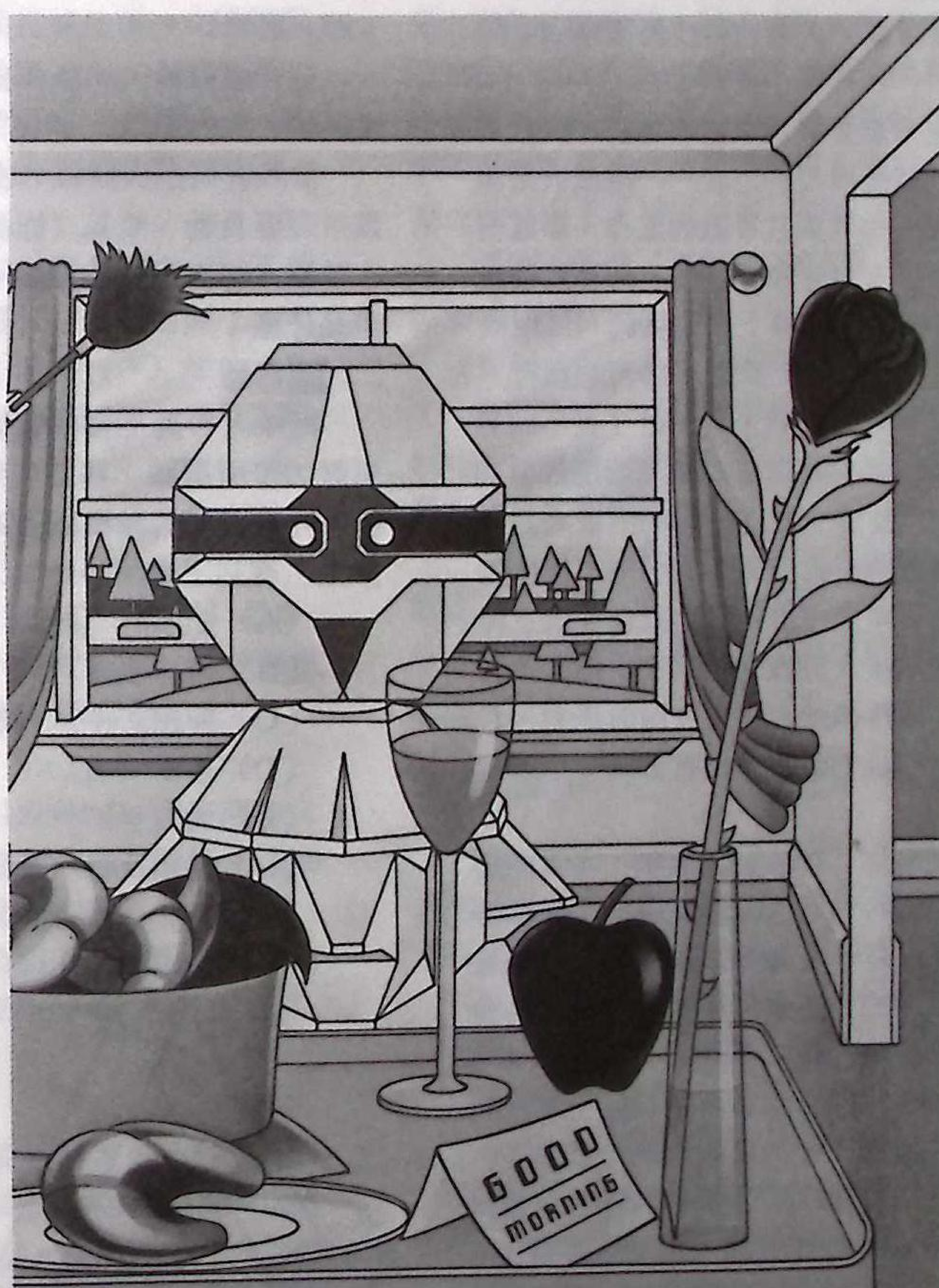
等等，數目之多皆因為不同的發展者用不同的語言去控制製成品，機械人語言大概不會有一個標準，要與一個機械人溝通可真不是一件易事。

機械人套件製作

套件式的機械人現在已普遍有售，由最簡單的模型玩具，到現時本港流行的"無比敵"以至英國出品的單臂機械人等等，皆可經由正途購得的，在英國有機械人套件郵購，機械人名為 HEBOT Ⅱ，它可以做些簡單的工作，如前、後、左、右、眨眼、轉彎等等，而命令則由一部EX 81微電腦所控制，整個套件售價只不過合港幣 900 元左右，零售商之地址如下，有興趣的讀者可以去信查問。

POWERTRAN
(CYBERNETIC DIVISION)
PORTWAY INDUSTRIAL ESTATE
ANDOVER HANTS SP10 3WN
ENGLAND □

淺白語言控制系統（PLAIN LANGUAGE CONTROL SYSTEM）的成果，上文所述機械人的功能成為事實為期不遠。記憶量價格日趨廉宜，口語認知，感應系統等等條件下，機械人的發展很樂觀。□



機械人的基本認識

讀者家裏可有一個、兩個，甚至三個的機械人？這個問題相信爲期不遠了，或許是五年、十年，更悲觀的展望也不出廿年，機械人時代便來臨，就如現在很多人的家裏也擁有微型電腦一樣的普遍。

假若問你到底機械人是怎麼樣的？你一定會答其外貌似人形，但由機械推動，並擁有一個小腦袋。不錯，這就是我們腦海中及科學幻想小說中的模樣，但直至目前，人類的科技仍未達到這點。

在本篇要談及機械人之前，讓我們先給一個機械人的定義——機械人是專責執行一些危險且單調的工作，而這些工作利用人力會過於浪廢，但在某程度上需要技巧及智慧；因此我們可以說機械人（ROBOT）只是一部可以執行一連串程序的指令的工具，這工具不同於電腦，因爲它可以直接使其環境四週的現實世界改變。較高一等的機械人是可以對四週的事物作簡單的感應，再依照預藏的資料以作出相對的動作。第三種的有關機械人的概念是眞正來自人工智能的，這種機械人可以有識別能力，且可以對相關的事物作出有系統的思想聯系。

使用到「智能」兩個字，不其然會令人勾出科學幻想電影或小說中的惡夢，機械人是一隻擁有無比力量，智慧及暴跳如雷的一隻鋼鐵怪物，他們甚至會支配或根除人類，亞斯武夫先生（A—SIMOV）唯恐此類事件發生，首先制定出三條人工智能機械人的法則：

一）機械人不會傷害人類，就算是活動時亦不容許令致人類受到損害。

二）機械人一定要服從人類的命令，除非命令是違背了第一項法則。

三）機械人必須自我保護，以求儘可能生存最長的時間，但這種保護必須不會與第一及第二項法則有所衝突。

由於機械人需要有複雜的心智去單獨進行工作，因此要它接受這種法則，實在並不容易，所以必須要向機械人灌輸一套道德標準。製造這套標準涉及它們在社會上的權利，故此是很複雜艱巨而重要的課題。

機械人如何工作？

人類做日常工作，主要分爲三個層面：

一）某種程度的移動，簡單如郁動我們的器官和肢體（脚、眼等）或轉移我們的位置（如走路、跑等）。

二）輸入及輸出資訊（說話、觸摸、聽、看等）。

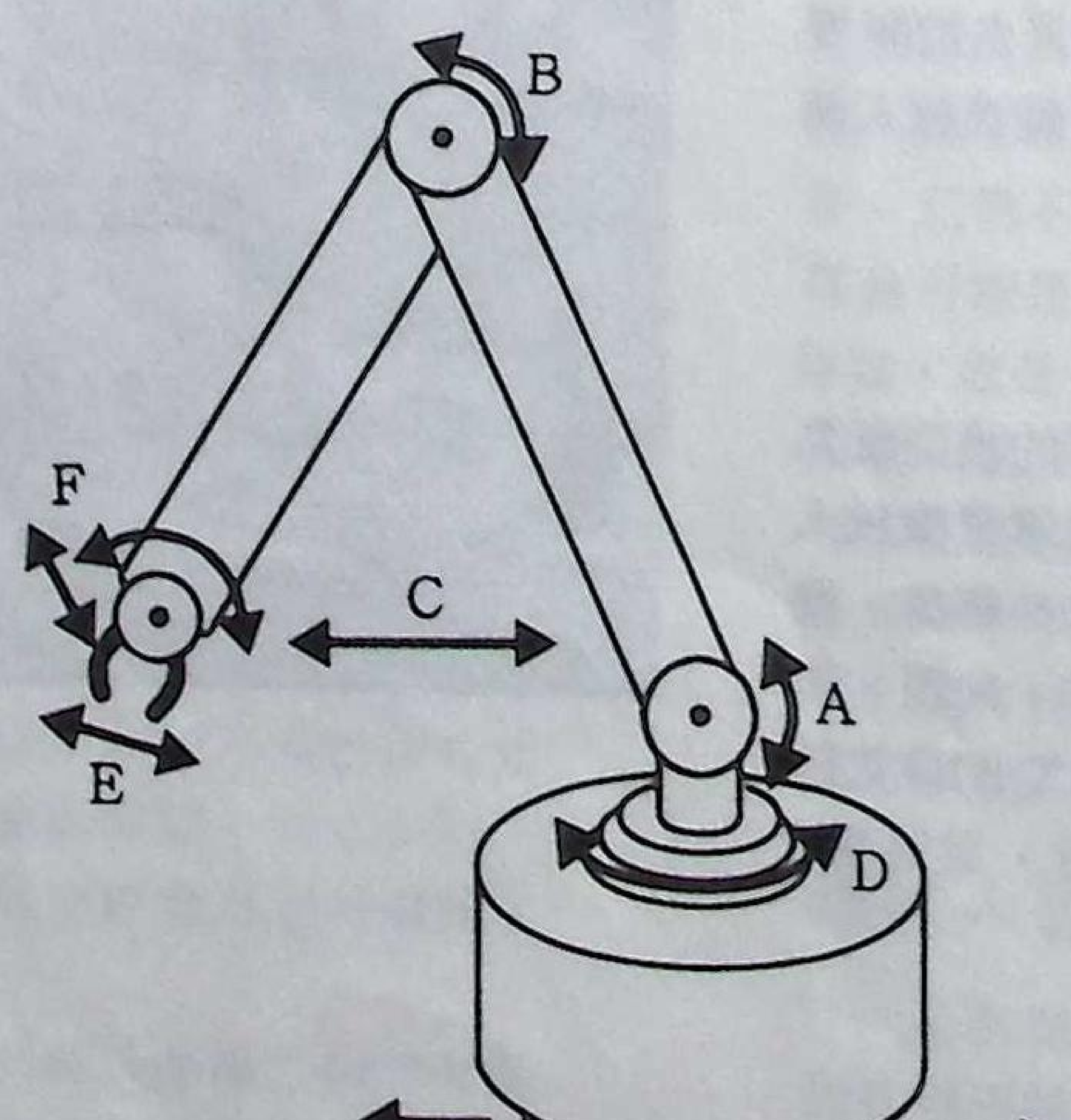
三）感應輸入然後作出反應（思考決定）。

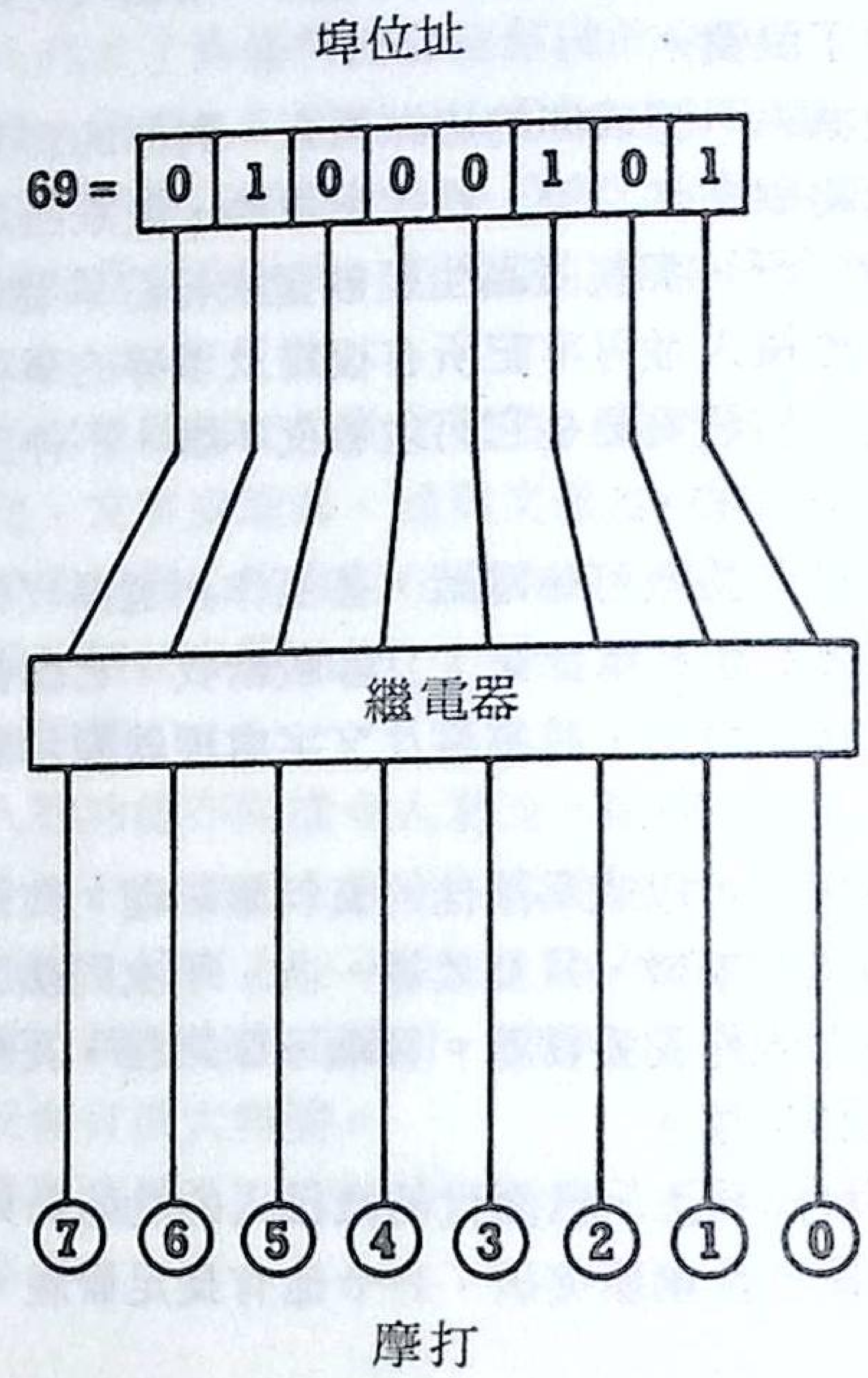
當然這只是很簡單的模式，我們人類亦需要食物、氧氣（能源）、熱和光。機械人亦有同類的需要，視乎他的功能是什麼：

■移動

機械人移動主要依賴水壓，摩打或氣體力學所驅動，移動的形式可以由開關掣所控制，例如機械人的臂多用於工業上，有以下幾種移動形式：

（A）上或落（肩）
（B）彎（手肸）
（C）伸出／伸入（肩與手肸）。
（D）搖動（肩）
（E）鉗（手）
（F）搖動及鉗（腰）





每一個接合點都可以獨立移動，以務求手臂可進行複雜的操作，透過連串的電驛，電腦操縱每一個磨打的獨立開關掣。電驛均連接上電腦記憶某一特定位置，而負責控制的程式將二進制的數據號碼放置於該位置上。以8 BIT的微處理器爲例，任何放在該位置的數字均是0及1的組合。八個電驛連接用家埠，每個受一個BIT所控制。如果某一BIT是1，摩打就會作出反應而開掣，如果位元是 ，磨打就會閉掣。

舉例，我們將69這個數字放在記憶位置，0，2及6摩打便會開動。如果要將磨打關上，控制程式就將0放在位置上。

更複雜的程式，可令各類滑輪或行動系統配合，令機械人可隨處移動。這種技術，必先要控制摩打驅使機械人可以行前、退後，或側身而過。

■輸入及輸出

需要輸入或輸出的資訊視乎機械人目的，選擇有三個，例如和人類有五個感覺：視覺、嗅覺、味覺、觸覺和聽覺。

●視覺——排列感光細胞可以顯示某物體兩度空間的形狀。有光令細胞製造電子輸出，而由電腦翻譯成二進數據

的1，同時，如因物體阻碍了光綫，使細胞感不到光，所產生的電子輸出，變成二進數據的0，二進制的排列就代表了物體的形狀。更爲複雜的視覺資訊輸入是透過視感鏡頭，圖畫可立即化爲數據而儲存。這種方法多用於工業生產綫上，以測試不符規格的產品。

●嗅覺及味覺——這些過程是以複雜的化學分析爲基礎。英國利蘭車廠用這概念製造機械臂去檢查新車的裂痕。檢查步驟首先將新車注滿了氦氣，機械臂就會檢查外殼及窗門的接口，一旦感應了氦氣，便會向控制單位發出電子訊號。

●觸覺——壓力及張力轉換器，可以將物理能量化爲電力訊號，而由處理器解碼。這些訊號可提供更詳盡細微的資訊，使瞭解壓力與張力的程度。

●聽覺——聲音的頻率透過耳筒化爲數據而發出訊號。無論任何聲音甚至說話語言均可以用二進形式儲存。這種技術已應用於航空上，飛機的電腦可以明白飛機師的口頭指令，而作出相應行動。

事實上，另外還有兩個人類感覺，不過通常被人忽略：

●內在意識感應（PROPRIOCE－PTION）——自我感覺到軀體肢體的位置，例如，就算我們閉上雙眼，仍可以將雙手互相接觸。

●平衡（BALANCE）——有關地心吸力的複雜關係，雖然這兩項感應加入機械人身上似乎很困難，但事實上並不如此，例如平衡方面，可以用陀螺平衡儀做到。

機械人亦可以做到有人類天賦的大部份的感應和觸覺，只要加上雷達，紫外光，紅內綫，聲納，超聲波接收資料，輻射，磁場及微波。

明顯地，同樣的科技，亦可能令機械人輸出資料，只要能夠令人類或其他機器明白便可。

機械人能做決定是非常重要的，它感應了輸入後，可產生一連串有可能性選擇的輸出。輸出本身並可影响緊接而來的輸入。

這自我調節系統稱爲FEEDBACK CONTROL LOOP(回饋控制迴路)。簡單的示範例子是自動交通燈控制系統：

1）在十字路口排長龍的汽車數目由馬路上感應器探測出來。

2）根據所獲得的資訊，處理器會改變燈號，以避免任何一個方向的交通大爲擠塞。

3）當車龍數目減少後，交通號又會變爲紅色，而又開始另一次的計算。

紅綠燈初期使用時，曾被稱爲「機械警察」。原理上與我們所說的機械人一樣。

機械人能夠作出思考決定，只不過是簡單數學上運算，有三個選擇：（1）大過；（2）小過或（3）等如。

以交通燈的例子來解釋其實是三項算式：

10 INPUT CARS（輸入車輛數目）

20 IF CARS＞30 THEN CH－ANGE LIGHTS（如果車輛數目多於三十部，即轉變燈號）

30 GOTO 10

這項算式，只是控制一個方向的交通流量，如果是四通八達的路口，那麼就要數項算式，同一時間運算。

而有關形狀及聲音等感應輸入，比較之下則困難得多。

文字認知

表面看來，要電腦將輸入的二進模式與儲存於記憶系統中的模式互相比較，是輕而易舉的事。當電腦於一九五〇年代面世時，電腦專家均豐有信心認爲可以用ＡＩ令電腦有認知效能。但在實際上，並非如想像那般容易，問題出現於處理需時，儲存及比較大型及複雜模式上，電腦亦難分辨差別少許的模式。我們用家庭電腦亦可重複這個試驗，目前已有現成的配件可以容許你以語言作爲輸入形式，製作簡單的程式教導電腦認識幾個口語文字。

其中一個ＡＩ研究是以ROSEN－BLATT PERCEPTION MODEL爲基礎。用〝COMMITTEE〞原則，去令機械人有分析能力。每個〝COM－MITTEE〞（委員會）的成員均是程式內分隔獨立的例行程序，逐一去與輸入的模式比較。每一個成員都有其與其他不同的特徵，而會去選擇適當的模式。程式會收集所有成員的資料，達致過半數的決定，並將結果通知使用者。如果決定是錯誤的話，程式會要求使用者通知它，應該採用那個選擇。程式接着會改變有錯誤的委員會的特徵，這過程不斷重復直至大多數成員獲得正確的決定。此情況下，程式便算〝學懂了〞。

舉例：教導〝BIG EARS〞去認知〝ONE〞和〝TWO〞，如果程式委員會有八個成員，當認知〝ONE〞時，三個成員對，五個成員錯，那麼大多數的決定就變成錯誤。程式會再更改五個答錯成員的特徵。

同樣類似的分析原則是利用快速平衡處理，稱爲〝WIZARD〞，它可以分辨每張不同的人類臉孔，甚至更可說出臉孔是否正在微笑。

未來發展

我們都會很有興趣，想知道機械人將來是否會成爲家庭一份子，好似目前的洗衣機，微波焗爐，自動錄影機一樣，答案是機會很大。預料，未來家庭電子用具將由中央電腦控制。這個完整的生活環境，無疑是〝智能〞居所，未來二十年內，我們對機械人會愈來愈熟悉，機械人是否會用於節省人力或取代人力的工具，要視乎科技社會如何解答很多道德上的問題。機械人究竟可以與人相似到何程度？在於我們的選擇。■

完整 CATALOG

本工具程式能夠將磁碟上所有 FILE 的名字印出，包括經已被 DELETE 的 FILE。在必要時，我們可救回這些被"意外"刪掉的 FILE。

本程式的用處可能並不太明顯，但有時你意外地 DELETE 一個 FILE 而又好想救回，程式便有用了。雖然你可以用其他" RESTORE PROGRAM" 去救回 FILE，但可能你完全不知道究竟有幾多 FILE 是意外地被 DELETE。

這個程式也可以印出這個 FILE 的種類（FILE TYPE），和LOCK 或UNLOCK FILE。所有被 DELETE 的 FILE 會以特別的方法顯示在螢幕上。

```
1  REM   ********************
2  REM   *                  *
3  REM   *   FULL CATALOG   *
4  REM   *                  *
5  REM   ********************
10  LET D$ =  CHR$ (4)
20  PRINT D$;"BLOAD RWTS"
30  TEXT : HOME : VTAB 4
40  PRINT "THIS PROGRAM WILL DISP
   LAY ALL OF THE FILES THAT AR
   E ON YOUR DISK ";: INVERSE :
    PRINT "INCLUDING": NORMAL
50  PRINT "THOSE FILES WHICH HAVE
    BEEN DELETED AND HAVE NOT B
```

本程式在程式磁碟上的 FILENAME 是 FULL CATALOG,RWTS

任意定義鍵盤

這個短短的程式，可使你任意重新安排鍵盤上的字鍵。用法非常簡單，RUN 程式後，打入要改動的字鍵，然後再鍵入代替的字鍵即可。假如你喜歡，可將所有按鍵全變爲A或其他。

本程式的唯一缺點，就是只有在程式 RUN 時，改動才能生效，若要同時應用於他工作便不能了。本程式在程式磁碟中的 FILENAME 是 DEFINE KEY。

編者按：今期我們出的題目就是希望讀者能編寫出一個程式，可以任意將鍵盤上任何一個鍵，而且可加以應用。例如本刊第五期「16進制數字鍵盤」及第一期「突破傳統的鍵盤」，程式執行後，可以加以利用。各位不妨參考其中的做法。

我們樂意送出十二期雜誌贈閱給三位將程式設計得最好的讀者（例如得獎者經已訂戶，則可於訂閱期滿後再有十二期雜誌。）

鑑於第五期刊出之光速計數器及問題棋盤，讀者們反應非常熱烈，而且有些答案或修訂非常有水準，唯獎額有限，使我們取捨十分困難。我們唯有依據清式設計的技巧，及說明文件的詳細程度來決定獲獎者。

假如參賽的程式有一定長度，敬請連同錄音帶或磁碟一併寄回或送來我們的讀者服務部（當然，這並非必要），無論得獎與否，我們都會退回給參加者。我們將會優先審核有磁帶或磁碟附寄來的參賽程式。

參賽作品請寄九龍中央郵政信箱 C.P.O. 71193。電腦時代出版社收。信封面請註明「任意鍵盤」。

```
130  TEXT : HOME : CLEAR
140 X = 1:Q$ =  CHR$ (34)
150  DIM A$(96),Z$(96)
160  VTAB 1: HTAB 1: PRINT "TO ST
     OP...PRESS<RETURN>WITH NO EN
     TRY"
170  VTAB 5: HTAB 1: CALL  - 958:
      PRINT "WHAT KEY WOULD YOU L
     IKE TO CHANGE?";: GET A$(X):
      INVERSE : PRINT A$(X): NORMAL
180  IF A$(X) =  CHR$ (13) THEN 2
     50
190  PRINT : PRINT "CHANGE IT TO
     WHAT ?";: GET Z$(X):: INVERSE
     : PRINT Z$(X): NORMAL
200  PRINT : HTAB 12: PRINT Q$;A$
     (X);Q$;"IS TO BE ";Q$;Z$(X);
     Q$;
210  PRINT : PRINT : PRINT "IS TH
     IS CORRECT?";: GET V$
220  IF V$ <  > "Y" THEN 170
230 X = X + 1
240  GOTO 170
250  HOME
260  PRINT "TYPE ANYTHING...PRESS
     <ESC> TO QUIT...": PRINT : PRINT
270  GET L$
280  IF L$ =  CHR$ (27) THEN 340
290  FOR A = 1 TO X
300  IF A$(A) = L$ THEN  PRINT Z$
     (A);:A = X: GOTO 330
310  NEXT A
320  PRINT L$;
330  GOTO 270
340  VTAB 22: HTAB 1: END
```

```
RWTS                      0391-    03
                          0392-    00
0380-    A9 03            0393-    20 00 00
0382-    A0 8A            0396-    01 00
0384-    20 D9 03         0398-    00
0387-    60               0399-    60
0388-    00               039A-    01 00
0389-    00               039C-    01 EF
038A-    01 60            039E-    D8
038C-    01 00            *L
038E-    00
038F-    00               039F-    76 FF
0390-    9B               03A1-    1B
```

```
    EEN WRITTEN OVER.": VTAB 22:
     PRINT "          **  FULL C
    ATALOG  **         "
60  VTAB 12: PRINT "PLEASE INDICA
    TE WHICH DRIVE CONTAINS THE
    DESIRED DISK (DEFAULT=1) ";:
     GET R$
70  IF  VAL (R$) = 1 THEN  POKE 9
    08,1: GOTO 100
80  IF  VAL (R$) = 2 THEN  POKE 9
    08,2:GOTO100
90  IF  VAL (R$) < 0 OR  VAL (R$)
     > 2 THEN  PRINT : PRINT : PRINT
    : INVERSE : PRINT "PLEASE RE
    MEMBER THAT YOU CAN HAVE AT
    ": PRINT "MOST 2 DRIVES CONN
    ECTED TO ONE CARD!!": NORMAL
    : FOR P = 1 TO 2500: NEXT P:
     VTAB 10: HTAB 1: CALL  - 95
    8: GOTO 60
95  POKE 908,1: REM  DEFAULT DRIV
    E NUMBER
100 ITRK = 910:ISECT = 911:ICMD =
    918:IBUFP = 915
110  GOTO 200
120  REM
130  REM  *** READING SUBROUTINE*
    **
140  REM
150 BUFP = 32:SECT = 15: POKE ITR
    K,17: POKE ICMD,CMD
160  POKE ISECT,SECT: POKE IBUFP,
    BUFP
170  CALL 896
180 SECT = SECT - 1:BUFP = BUFP +
    1: IF SECT > = 1 THEN 160
190  RETURN
200  HOME : VTAB 2: HTAB 6: INVERSE
    : PRINT " READING DISKETTE D
    IRECTORY ": NORMAL : POKE 34
    ,3
210  PRINT
220 CMD = 1: GOSUB 150
230  VTAB 2: HTAB 1: CALL  - 868:
     HTAB 11: INVERSE : PRINT "F
    ULL     CATALOG ": NORMAL : VTAB
    4
240  REM
250 A = 8203:B = 1:A1 = A
260  IF  PEEK (A) = 0 THEN 420
270  IF  PEEK (A) = 255 THEN 340
280 CNAME$ = ""
290 C = A + 2: FOR K = 1 TO 30:D =
    C + K
300 CNAME$ = CNAME$ +  CHR$ ( PEEK
    (D))
310  NEXT K: GOSUB 440: PRINT CNA
    ME$: LET Q = Q + 1: IF Q / 1
    9 =  INT (Q / 19) THEN  POKE
     - 16368,0: GET R$
320 A = A + 35:B = B + 1: IF B >
    7 THEN A1 = A1 + 256:A = A1:
    B = 1
330  GOTO 260
340 DNAME$ = ""
350 C = A + 2: FOR K = 1 TO 29:D =
    C + K
360 DNAME$ = DNAME$ +  CHR$ ( PEEK
    (D))
370  NEXT K
380  INVERSE : GOSUB 440: PRINT D
    NAME$: LET Q = Q + 1: IF Q /
    19 =  INT (Q / 19) THEN  POKE
     - 16368,0: GET R$
390  NORMAL
400 A = A + 35:B = B + 1: IF B >
    7 THEN A1 = A1 + 256:A = A1:
    B = 1
410  GOTO 260
420  TEXT
430  END
440  REM
450  REM  UNLOCKED FILE TYPE
460  REM
470  LET FT =  PEEK (C)
480  IF FT > = 128 THEN  GOSUB 5
    60: RETURN
490  ON FT GOTO 510,520,530,540
500  PRINT " T ";: GOTO 550
510  PRINT " I ";: GOTO 550
520  PRINT " A ";: GOTO 550
530  PRINT "NEW FILE TYPE "
540  PRINT " B ";
550  RETURN
560  REM
570  REM   LOCKED  FILE TYPE
580  REM
590  ON FT - 127 GOTO 610,620,630
600  PRINT "*B ";: GOTO 640
610  PRINT "*T ";: GOTO 640
620  PRINT "*I ";: GOTO 640
630  PRINT "*A ";
640  RETURN
```

■

CP/M 3
入門基礎篇

檔案管理

黃偉明

電腦操作系統是所有應用程式的工作核心；而檔案管理系統是電腦操作系統的支柱。所謂檔案管理系統是提供用家一系列的程式，可把資料或程式有系統地儲在後備存儲庫（BACKING STORE）——磁碟或磁帶等，給它們名字，和其他讀、寫或更改等服務。由於電腦的應用需要大量的程式和資料，它們需要後備存儲庫儲藏，所以資料庫和檔案的管理問題是操作系統的重心。

一個完善的檔案管理系統必須包括以下要點：

一、設立一個統一和標準的檔案名稱法則。

二、提供一系列檔案操縱和檔案錄工作服務程式。如檔案的開啓、改名和消除等。

三、提供不同讀寫方法的服務程式。

四、能適應不同的後備存儲庫類型和容量，用家能由自地選擇所需要的型號和大小。

五、完善的檔案保安系統，不致重要資料被未經許可的人抄錄或更改。

六、提供重要檔案的保險（BACKUP）方法，避免或減輕因爲意外或不小心所引致的損失。

七、完善和效率高的後備存儲庫空間安排和管理。

但不同的電腦操作系統將因應不同的重點和要求而有所取捨。

現在可嘗試根據以上各點來分析CP／M的檔案管理系統。CP／M所用的檔案名稱法則和DIR指令已在上期詳述；檔案的名稱以不超過八個字母或數字代表，而類別以三個或以下字母或數字代表。要注意的是所有在CCP（即輸入指令處理程式）下使用的檔案名稱和類別所用字母必須是「大楷」，名稱中含有「小楷」英文字母的只能在其他程式中使用；所以應盡量以「正楷」字母作爲檔案名稱。

CP／M提供了大約十六個服務程式協助其他程式担任檔案操縱及磁碟讀寫等工作，全部放在BDOS(基本磁碟操作程式）中。由於CP／M以磁碟作主要後備存儲庫，所以沒有協助磁帶讀寫工作的服務程式。至於服務程式的使用和詳細意義不在本文範圍內。

由於CP／M是微電腦操作系統，即使用的後備存儲系統都是中小型的磁碟機，包括軟磁碟機和硬磁碟機：

一、軟磁碟機的種類也很多，主要根據以下各點的組合而有所分別：

A）磁碟的直徑：——主要分爲五吋和八吋兩種。八吋磁碟的容量較大，適合實務應用的電腦系統；但五吋磁碟價錢較廉，適合家庭用電腦系統。

B）單面和兩面用磁碟機：——兩面用磁碟機需要有兩個讀寫器，所以製作比較複雜和嚴謹，價錢昂貴而且少人用；所用的磁碟也要較高質素。

C）磁碟的密度：——分爲普通和高密度兩種。普通密度是指3200BPI(BITS PER INCH）的磁碟；高密度是指6400BPI的磁碟。高密度磁碟可當作普通密度磁碟用。

D）磁碟扇區容量：——磁碟扇區的容量主要是由磁碟機，所用介面電路和軟件決定的，分爲128、256、512或1024單元（BYTE）。

E）軟性和硬性扇區分割：——軟性扇區分割即利用程式把磁碟上的資料磁軌等，分爲若干扇區來方便磁碟讀寫工作；而硬性扇區分割是利用磁碟近中心處的小孔，同樣把磁碟分割。由於軟性分割能分割出不同大小的扇區容量；而硬性分割由磁碟機決定，不能更改，所以軟性分割較通用。

二、硬磁碟機：——和軟磁碟機大致相同，分別在於它的轉動速度快很多而且容量大，由1MB到2500MB不等。用家若考慮用硬磁碟機，必須先詳細估計所需容量。

當用家選定所用的磁碟機後，應選定磁碟中存儲檔案的詳細組織。通常CP／M用的磁碟的第一或二條資料磁軌（即最外邊的）是保留用作系統起動。接着的磁軌分成扇區；每扇區代表一個或以上的資料節（RECORD），每資料節都有一個節號；節號由外到內的增加。而幾個資料節又成爲一個資料節組（BLOCK）；節組號也是由外到內地增加。最首的幾個資料節組是被用作檔案錄。一般的CP／M系統，每個檔案在檔案錄中佔上三十二單元（20HB）。

當磁碟組織選定了後，便由系統程式員把詳細資料加在系統程式中的磁碟組織節組（DISK PARAMETER BLOCK）中；把磁碟格式化（FORMATE）好，然後將新的CP／M寫在碟上，再把工具及應用程式抄上。

CP／M檔案系統中，有一個工具程式——STAT、COM。首先，我們利用它來看看正用著的磁碟組織，請依照附例打下「STAT A：DSK：」便能看到A——磁碟機所採用的磁碟組織。附例中使用的是一個五吋直徑、單面的普通密度磁碟機。STAT 顯示出磁碟上有三條綫軌保留作CP／M起動（一）；（二）其餘的每條綫軌平分成三十二分割節；（三）總共有一千零二十四隻資料節，而每資料節的容量是一百二十八單元；（四）所以磁碟的總容量是一百二十八單元（128KB）；（五）由於每八隻資料節合成一隻資料節組，而用了兩隻資料節組作爲磁碟的檔案錄；（六）所以最多可容四十八個三十二單元的檔案錄資料（EN—TRIES）。每個檔案錄資料中，有十六單元用來記下某檔案所佔有的資料節組，共十六個資料節組或一百二十八個資料節；（七）稱之爲一個檔案節（EXTENT）。若檔案需要超過一百二十八個資料節，它可在檔案錄上多佔一些檔案錄資料來記錄它第二個或以上的檔案節和所佔的資料節組。

附例中把同一磁碟所用的磁碟組織節組列出以供參考，因爲 STAT 所列出的磁碟特點是根據這節組混算出來。至於磁碟組織節組將不作詳細討論。■

（ 未完　下期續 ）

```
STAT A:DSK:

     A: Drive Characteristics
  1024: 128 Byte Record Capacity ——————————— 3
   128: Kilobyte Drive  Capacity ——————————— 4
    48: 32  Byte Directory Entries ——┐
    48: Checked  Directory Entries ——┘——————— 6
   128: Records/ Extent ———————————————————— 7
     8: Records/ Block ————————————————————— 5
    32: Sectors/ Track ————————————————————— 2
     3: Reserved Tracks ———————————————————— 1


A>XDPB
Disk Parameter Block for drive A
SPT: 0020 records per track
BSH: 03  recno >> BSH  =  block number
BLM: 07  recno AND BLM = record in block
EXM: 00  logical extent versus physical
DSM: 007F highest block number (origin 0)
DRM: 002F hignest directory number (origin 0)
ALV: C000 bits reserving directory blocks
CKS: 000C SIZE OF CHECK VECTOR IN BYTES
OFF: 0003 NUMBER OF RESERVED TRACKS
DA93: 20 00 03 07 00 7F 00 2F 00 C0 00 0C 00 03 00     ....../.@.....
```

■

捉蟲訓練——閃動訊息

本刊自第五期增加「捉蟲訓練」一欄，所得反應非常良好。「問題棋盤」刊出後，收到接近百封參加信，大部份都能找到問題所在。令我們非常高興的是讀者們對 BASIC 語言的能力有一定的水平。

今期的捉蟲訓練是有關一個能令一段訊息在螢幕上任何水平位置由右至左的不停重覆移動。

程式中，變數 S 是訊息移動的速度，數值越細，速度越快，而 V 則是螢幕水平位置。

這個程式本身不會出現任何 SYN—TAX ERR ，它能運行，不過結果當然不是我們想要的。聰明的讀者請將正確的答案寄來給我們。我們將會送出一張程式磁碟給最佳答案的一位讀者。答案將在下期揭曉。

請連同姓名、地址、聯絡電話寄來九龍中央郵政信箱 71193 號。信封面請註明「捉蟲」。

同時，我們極之歡迎讀者們自己設計一些類似的捉蟲訓練題目投寄給我們。程式長度沒有限制，當然越精簡越好，程式以 APPLESOFT 編寫，並需符合下列各項要求。

1、程式必須是 APPLESOFT ，任何 POKE 機械語言的子程式或 CALL 該子程式必須正確。
2、程式必須能出現有某些結果，而不會出岔子，換言之絕不能有 SYN—TAX ERROR出現。
3、程式必須附加簡短說明。
4、必須連同答案寄來。

假如閣下投寄來的捉蟲訓練程式是有水準，我們將贈閱十二期雜誌給你，或者再多一年的免費讀訂。投稿請在信封面寫上「投稿捉蟲」。

```
]
]LIST

10  GOSUB 200
20  FOR H = 40 TO 1 STEP  - 2
30  VTAB V: HTAB H:C = C + 1
40  IF C < L THEN  PRINT  MID$ (A
    $,1,C)
50  IF C = L OR C > L THEN  PRINT
    A$
60  FOR D = 0 TO S: NEXT D: NEXT
    H
70  FOR H = 2 TO K:C = C + 1: VTAB
    V
80  IF C = L OR C < L THEN  PRINT
    MID$ (A$,H,C)
90  IF C > L THEN  PRINT  MID$ (A
    $,H,L)
100 FOR D = 0 TO S: NEXT D: NEXT
    H
110 KEY =  PEEK ( - 16384): ON KE
    Y > 128 GOTO 240
120 GOTO 20
200 CALL  - 936
210 VTAB 5: PRINT "TYPE IN YOUR
    MESSAGE,PLEASE:": INPUT A$
220 S = 80:V = 21
230 A$ = A$ + " ":L =  LEN (A$): RETURN
240 END
```

■

姚克林

APPLE PASCAL專欄

啟動方法與程式結構

上期講及 APPLE PASCAL 有四隻磁碟，即 APPL—E0，APPLE 1，APPLE 2 及 APPLE 3。如果大家是用一部磁碟機操作，以下便是啓動的步驟。

一、把你的 APPLE 的電源關上，接着把 APPLE 3 磁碟插入磁碟機。

二、把電腦電源打開，過幾秒鐘後，把 APPLE 3 磁碟拿掉，並且把 APPLE 0 磁碟放進磁碟機裡去。

三、按下〝RESET〞鍵，大約過十數秒，螢光幕上就會出現一些訊息。最後螢光幕的最上端就會出現一列 COM—MAND 訊息。

COMMAND訊息一出現，就表示你已把 PASCAL 啓動了。大家要注意，如果你沒有使用 80 字行咭，你看到在螢光幕上的只是 COMMAND 訊息的一部份，而另一部份只需按下〝CTRL A〞鍵，就會出現於螢光幕上。再按一下〝CTRL A〞鍵，便會回復到原來的左半部上了。

在未教大家怎樣運用 COMMAND 提示列前。在此先談談 PASCAL 程式與 BASIC 的程式有甚麽分別。以下是一個最簡單 PASCAL 程式的例子。

```
BEGIN
            END.
```

這個程式中包含兩個PASCAL 語言中的保留字，BE—GIN，END和END 後面的句號，不要看輕這個句號。如果遺漏了它，這個程式在編譯時就會出現毛病。當然大家知道這個程式是完全沒有用的。但這裡可以導出 PASCAL 程式的最主要骨幹。PASCAL 只會執行介乎 BEGIN 和 END 之間的程式，由最高一行執行至最低一行，現在讓大家看看另一個例子。

```
PROGRAM ADDITION;
BEGIN
WRITE(238+429)
END.
```

這個例子中較上一個例子多出兩列字。這就是 PROG—RAM ADDITION ；和 WRITE（238＋429）。在未進一步解釋整個程式的意義時，大家可以自我猜度一下這個程式執行起來時有甚麽意義。在螢光幕上出現甚麽呢？答案是 667 對嗎。現在讓詳細解釋整個程式的內容。

PROGRAM ADDITION ；PROGRAM的後面必須要有一個空格，而 ADDITION 是這個程式名稱，而 ADDITION 的後面是個分號。因此大家知道這程式名稱叫 ADDITION 。程式的名稱沒有特別限制，任何長度的字串都可使用，惟中間不能留有空格。

BEGIN意即程式開始執行。

WRITE（238＋429）意即將括號內的運算執行並將答案 667 輸出螢光幕上。WRITE一字是保留字眼。跟著其後總是有一對括號。 WRITE 一字的意思是將括號內所執行的結果輸出螢光幕上。

END.相信還記得 END 字的意思是程式執行完畢。而 END 字後面不可忘記有一句號。

如果大家對上面的程式沒有問題的話，相信大家都很心急想知道怎樣將上面的程式寫到磁碟上去，而親自執行一下這一個程式，大家不要著急，只需多數分鐘時間便可實現。

讓大家回到 PASCAL 啓動後，螢光幕出現 COMM—AND 一列認息之中，按F鍵，這時大家進入 FILER 層面，接着按一下N（NEW）鍵，意思是準備輸入一個新的程式，如果磁碟中已經有某個程式時，只需要把舊的內容刪去，好使新的內容能夠寫到上面去。這時電腦會問你是否想刪去舊的內容，此是打入Y鍵去回答這個問題。刪除舊內容後，電腦裏面建立了一片空白的區域，令大家可以將新內容寫進去。現在大家依然停留在 FILER 層面，在螢光幕最頂一行可以看見，要將新內容打入必須回到 COMMAND 層面。請按下Q鍵，現在你已經在 COMMAND 層面了，對嗎？

在 COMMAND 一列的提示列中，選出 EDIT 這一項目，按下E鍵，再按一下 RETURN 鍵便到達 EDIT 層面，這時再按一下I鍵，I是 INERT 的縮寫，到達 INERT層面後便可將新內容打入了。現在讓大家將上面程式打入吧。接着以 CTRL—C 的方法離開 INERT 層面，於是你已經成功地將新的內容輸入了工作區域了。在按下鍵後，螢光幕會出現5項選擇，請你按下U鍵，過一會大家便回到 COMMAND 層面。跟着是要將剛寫好的程式編譯一下，請按一下R（RUN）鍵，如果程式沒有出現問題，螢光幕便會出現執行的結果。如果程式出現問題，便會在編譯過程中宣示出來，指出錯誤的地方。這時大家需要再回到 EDIT 層面中作修改。

現給大家多舉一些例子，依照以上的步驟重新啓動，刪除舊內容後，打入下面程式……

```
PROGRAM GREETING;
BEGIN
WRITELN("GOOD MORNING.HOW ARE
YOU?");
WRITELN ("GOODBYE.THAT'S
ALL.")
END.
```

APPLE PASCAL

將以上的程式執行一下。
螢光幕是否出現：

```
GOOD MORNING.HOW ARE YOU ?
GOODBYE.THAT'S ALL.
```

現在重新打入上面程式，但是這次將 WRITELN 改寫成 WRITE ，接著把它執行一下，看看和剛剛的程式有甚麽分別。相信你現在明白 WRITE 與 WRITELN 的分別。

讓大家再做一個實驗，再把上面程式寫一次，這次把括號內的一對單引號刪去，再執行一下。這次在編譯時出現毛病了。

最後大家可能提出一個問題，在第一個例子當中，WRITE（238＋429）爲甚麽在編譯時沒有出現問題呢。這個問題只需要大家再將下面程式重新打入電腦，再經編譯程序所得的結果，便可解答大家的問題。

```
PROGRAM ADDITION;
BEGIN
    WRITE("238+429")
END.
```

原來電腦不會將在單引號內的數式運算，它只會將在單引號內的東西，完整地輸出螢光幕。一例 BASIC 中的 PRINT指令後「"　"」內的東西一樣。

相信大家對 PASCAL的使用啓動及程式的結構有一個初步的認識！下期我們再繼續討論其他。■

蘋果宣佈推出最新便携式 2C 型電腦

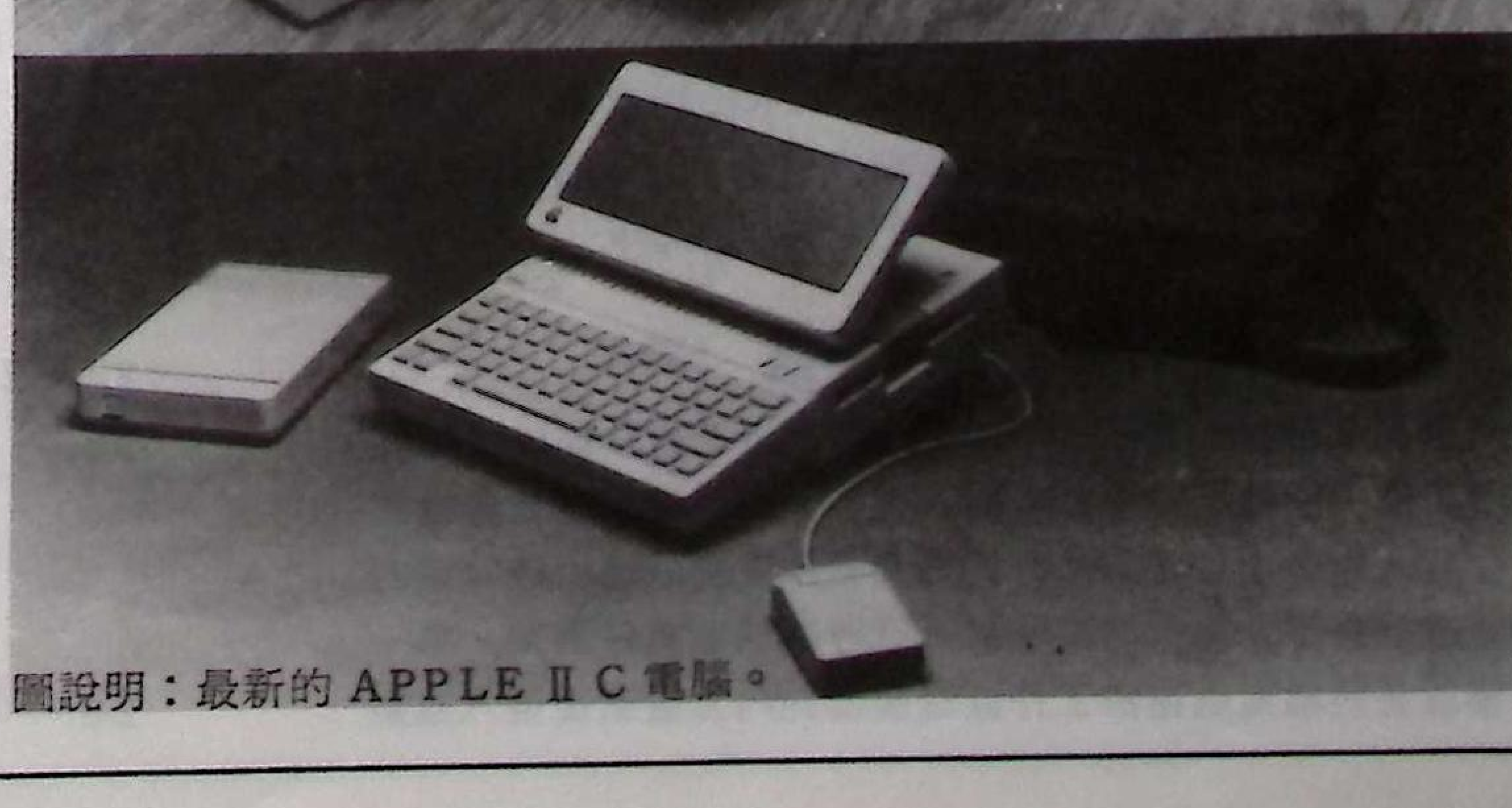
美國蘋果電腦上月（四月）宣佈推出一款便携式的 APPLE Ⅱ C型電腦，這款新電腦的設計完全可以適用蘋果電腦家族中最暢銷的 APPLE Ⅱ 私人電腦的各式軟件，但却較後者輕便得多。

這款全新的 APPLE Ⅱ C 型電腦推出後，將使 APPLE Ⅱ 系列的電腦買家今後有更多選擇。目前 APPLE Ⅱ E 及 APPLE Ⅲ 電腦所採用數以千計的各式軟件程式，同樣可適用於這款全新的 APPLE Ⅱ C 型電腦。據蘋果電腦本港總經銷太平洋行商業機器部發言人表示，這款新式的電腦已定於五月初在美國正式推出發售，但本港方面則仍未定何時推出銷售。

APPLE Ⅱ C 型新式電腦重量僅及三點四公斤（七點五磅）、具有 128 K BYTE 內藏式記憶容量，適合商業、教育等各種不同用途，亦能適合家庭電腦使用者的需要。

APPLE Ⅱ C 型電腦附有一具標準格式的鍵盤，單碟式驅動器，以及一個可隨意選擇購置的手動操作設計「老鼠仔」。□





圖說明：最新的 APPLE Ⅱ C 電腦。

DOS 3.3 專欄
甘貴祥

DOS 緩衝器

今期我們在這裡討論一下 DOS 緩衝器（BUFFER）和檔案經理器之間的關係，呼叫檔案處理器以保存記憶體時，須使用到一個 DOS 緩衝器。

這個 DOS 緩衝器含有檔案處理器會利用到的三個緩衝器（檔案處理器工作區，T／S（TRACK／SECTOR）表的區段，及資料區段），30 個 BIT 的檔案名稱緩衝器，及許多鏈結指標。整個 DOS 緩衝器共佔記憶體 595 B—YTE 。第一個 DOS 緩衝器的位址存在 DOS 的第一 BYTE 中（48K系統中爲 $9D00)。下一個 DOS 緩衝器的位址則存在第一個 DOS 緩衝器中，如此繼續下去形成一緩衝鏈（CHAIN)。其中最後一個緩衝器的鏈結（LINK）位址存零，若緩衝器未爲 DOS 所用，其檔案索引名稱欄的第一 BYTE 內存 $100，否則便爲開檔的名稱的第一字。滙編語言程式應依此規則運用，以免 DOS 重覆使用緩衝器。

檔案名稱須先存入緩衝器以取得緩衝器的唯一使用權，並於使用完畢後將 $00 存入檔案名稱的第一 BYTE 供 DOS 將來之用。若不如此，最後 DOS 會耗盡可用的緩衝器，甚至拒絕執行 CATALOG 命令。DOS 緩衝器格式如下：

DOS 緩衝器格式

BYTE	說明
000／0FF	資料區段緩衝器（長爲 256 BYTE）
100／1FF	T／S表區段緩衝器（長爲256 BYTE）
200／22C	檔案處理器的工作區緩衝器（長爲 45 BY—TE）
22D／24A	檔案名稱緩衝器（長爲 30 BYTE）

第一個 BIT 顯示出此 DOS 緩衝器是否正使用中，若第一個 BIT 內容爲 $00 表示此緩衝器 空着未用。

BYTE	說明
24B／24C	檔案處理器的工作區緩衝器之位址（低BY—TE／高 BYTE）
24D／24E	T/S 表區段的緩衝器位址。
24F／250	資料區段緩衝器位址。
251／252	緩衝器鏈的鏈結（LINK)，指向下一個緩衝器的檔案名稱欄之位址。若本緩衝器已是鏈中最末一個，則此處塡入零。

檔案處理器工作區

檔案處理器工作區含有許多變數，這些變數組成了檔案處理器在"開檔"時所須的所有資訊。每當檔案處理器做完一呼叫工作後，便將它所有的重要變數複製入使用者特定的檔案處理器工作區緩衝器。到下次呼叫檔案處理器時，第一件事就是將檔案處理器工作區緩衝器的內容重新複製入檔案處理器工作區的變數中，以再度處理上次呼叫之後離開的檔案。平常程式設計師不須爲此工作區的內容操心，因大多數有用的資訊都出現在參數表中。只是有時它可以使人對開檔案更加了解。以下爲檔案處理工作區的格式：

檔案處理器工作區之格式

BYTE	說明
00／01	檔案的第一個T／S表所在的軌道／區段。
02／03	檔案的現在（CURRENT）T/S 表所在的軌道／區段。
04	旗號： 80＝T／S 表之內容已更改，順重寫入磁碟中。 40＝資料緩衝器內容已更改，須重寫入磁碟中。 02＝ 冊的自由空間映圖（VOLUME FREESPACE MAP）已更改，須重寫入磁碟。
05／06	現在（CURRENT）資料區段所在的磁軌／區段。
07	目錄中檔案進入點的"區段偏移量"（SE—CTOR OFFSET）。
08	目錄區段中（07 所指的區段）檔案進入點的"BYTE偏移量"
09／0A	一個T／S表所能表示的最大資料區段數。
0B／0C	現在的T／S表中第一個區段的偏移量。
0D／0E	現在的T／S表中最後一個區段的偏移量。
0F／10	前一次所續的相關區段號數。
11／12	以" BYTE "爲單位的區段大小（256）。
13／14	相關區段群中現在的位置（現在指向那個區段）。
15	此區段中現在的 BYTE 偏移量。
16	未使用。
17／18	固定資料錄長度。
19／1A	現在資料錄號數。
1B／1C	指入現在資料錄的 BYTE 偏移量。
1D／1E	以"資料錄"爲單位的檔案的大小。
1F	本磁軌的下一個配置（ALLOCATE）的區段。
20	正受配置的現在磁軌。
23／24	本磁軌上可利用的區段的 BIT MAP。
25	檔案型式，00，01，02，04＝T（TE—XT)，I(INTEGER），A（APPLESOFT），B（BINARY）。（80 表已鎖制）。
26	槽號數乘 16 （如$60＝ 第6號槽）。

DOS 緩衝器

27　　磁碟機號數（01 或 02）。
28　　冊號數（以補數型式表示）。
29　　磁軌。
2A／2C　　未使用。

共通法則

使用 DOS 時，以下數組程式可供利用。

定置——FREE DOS 緩衝器：
以下副程式用來定置（LOCATE）一未配置的 DOS 緩衝器供檔案處理器使用。

```
FBUFF   LDA   $3D2      定置DOS 載入指標
        STA   $1
        LDY   #0
        STY   $0
GBUF0   LDA   ($0),Y    定置下一個 DOS 緩
                        衝器。
        PHA
        INY
        LDA   ($0),Y
        STA   $1
        PLA
        STA   $0
        BNE   GBVF      取得一個緩衝器。
        LDA   $1
        BEQ   NBUF      無自由緩衝器可用
GBUF    LDY   #0        取檔案名稱。
        LDA   ($0),Y
        BEQ   GOTBUF    是未使用的緩衝器。
        LDY   #36       已使用的緩衝器。
        BNE   GBUF0     去取下一個緩衝器。
GOTBUF  CLC             顯示已取得一個自由
                        的緩衝器。
        RTS             返回呼叫者。
NBVF    SEC             顯示未取得自由的緩
                        衝器。
        RTS             返回呼叫者。
```

判定正處於那一DOS 版本上：
有些程式含有與 DOS 版本有關的數碼，因此須事先檢查過 DOS 版。

```
CLC
LDA   #D        加$16BE於 DOS 載入指標。
ADC   #$BE
STA   $0
LDA   $302
ADC   #$16
STA   $1
LDY   #0
LDA   ($0),Y    取得 DOS 版本的號 數（2或3）
```

判定 DOS 是否在機器中：
以下指令須在呼叫 RWTS 或檔案處理器之前用之，以確定 DOS 已存入機器中。

```
LDA   $3D0      可 JMP 指令碼。
CMP   #$4C      檢查是否爲跳越指令。
BNE   NODOS     不是，DOS 仍未存入記憶體。
```

判斷選用的BASIC 種類：
許多程式的執行要靠 INTEGFR BASIC ROM或 APPLE—SOFT ROM 。以下程式可以檢查出正使用的BASIC種類，並可選所須的BASIC種類。首先載入一代表所須 BASIC 種類的數碼，INTEGER BASIC 爲 $20，APPLESOFT BASIC 爲$4C ，例如選用 APPLESOFT 時：

```
        LDA   #$4C      APPLESOFT 的
                        數碼。
        JSR   SETBSC    呼叫副程式。
        BNE   ERROR     語言尚未適用。
SETBSC  CMP   $E000     檢查是否爲正確的
                        BASIC。
        BEQ   RTS       是。
        STA   $C080     非，選擇ROM 咭。
        CMP   $E000     再檢查看是否取得正
                        確的 BASIC。
        BEQ   RTS       是。
        STA   $C081     否，再試 ROM咭。
        CMP   $E000     再檢查，現在如何？
RTS     RTS             無條件返回。
```

檢視是否正執行 BASIC程式：
用下列程式檢視 BASIC 程式是否正在進行，或是處於立即命令模式。

使用 INTEGER BASIC 時——

```
        LDA   $D9
        BMI   EXEC      程式正執行中。
        BPL   NOEXEC    程式未執行。
```

使用 APPLESOFT BASIC時——

```
        LDX   $76       取得鏈綫號數（LI
                        NK NUMBER）。
        INX
        BEQ   NOEXEC    程式未執行。
        LDX   $33       取得 PROMPT
        CPX   #$DD      PROMPT是“ ”
                        嗎？
        BEQ   NOEXEC    是，則程式未執行。
        BNE   EXEC      否，則程式正執行中
                        。
```

■

妙用無窮的128K RAM咭

本文詳細闡釋這咭的三大功能：

DOS擬磁碟機、CP/M擬磁碟機、PASCAL擬磁碟機

128K RAM 咭在市面上經已推出很久，可惜由於售價昂貴（超過半部6502 CPU 的價錢），所以未能如其他週邊咭（16K RAM咭、80字行咭等）受歡迎。而且大家對這張咭的特性及功用亦未能弄清楚。

雖然這張咭的售價是較其他咭貴上三、四倍（現時售價約由七百元至八百五十元左右），但這張咭的功能却是超乎我們想像。只要我們願意花點時間研究這張咭的使用說明，便可發現其中的奧秘。

128K RAM咭的功能可分三大部份：

① DOS 擬磁碟系統，它包括———

A、組成及安裝PSEUDO DISK

B、補充 DOS 功能

C、重新置放 DOS ，使主機記憶體空出 10.5K 空間以執行更大型的程式。

②PASCAL 擬磁碟系統，可配合APPLE PASCAL 1.1使用。

③CP／M擬磁碟系統。

DOS擬磁碟

用 DOS 擬磁碟，使用者可以像使用一般磁碟那樣的在128K RAM咭上存放程式。當然，128 K RAM 咭怎麼說也不像一個磁碟機，但是，配合適當的軟件，使用者用起來會覺得該咭像磁碟一樣。

爲了能把128K RAM咭當做擬磁碟系統使用，必須載入控制的軟件常式（CONTROLLING SOFTWARE ROUTINES），且必須完成一些初設（INITIALI－ZATION）步驟。DOS PSEUDO DISK套件程式由三個程式構成，以執行這種載入初設PSEUDO DISK的工作。這些檔案必須在同一磁碟上，才能組成及安裝PS—EUDO DISK。

一、PSEUDO DISK

目錄驅動器（MENU DRIVER）軟體程式，用以組成PSEUDO DISK 參數及安裝擬磁碟。

二、PSEUDO NRM

二進位程式，含有磁碟驅動器及 DOS 補充機能，以操作一個正常的 DOS （而非重新安放 DOS）。

三、PSEUDO MVD

二進位程式，由驅動器軟件及DOS 補充功能，以操作被重新安放過的DOS。

四、PSEUDO PARAMETERS

由PSEUDO DISK 程式的系統參數，位置與現有的記憶板種類等更新的檔案。

擬磁碟的要求條件及特性

擬磁碟係設計爲在標準及重新安置的DOS 3.2 或 3.3 上使用者。它可用在一張或多張 RAM 咭使用。擬磁碟最多可用到256 K 。它已具有一種能保留 RAM 咭空間供BASIC使用的能力，如果須要取存任何一種BASIC（在APPLEⅡPLUS爲APPLE SOFT；在APPLEⅡ爲INTEGER）。

要使用PSEUDO DISK 須經過兩個步驟：

（1）組成PSEUDO DISK

（2）安裝PSEUDO DISK

這兩個步驟必須依其先後次序進行。

組成PSEUDO DISK

此步驟主要是告訴系統記憶體電路板在那裏，以及如果只使用一部份的話，那一部份將被使用。要組成時，執行APPLE SOFT程式PSEUDO DISK、PSEUDO、NRM 及 PSEUDO MVD 必須與 PSEUDO 同在該磁碟上。

以下的索引目錄（MENU）會被顯示：

```
1. Look at pseudo disk set-up
2. Set-up pseudo disk
3. Install pseudo disk
4. Reconnect pseudo disk
5. Exit
```

選擇第2項目，則螢幕上會顯示：

```
SLOT 5      DRIVE 1    DIR      SECS 4
            BANK                BANK

SLOT      12345678     SLOT  12345678
0                      4
1                      5
2                      6
3                      7
-----------------------------------------
B=BASIC      Y=USED          N-NOT USED
-----------------------------------------
1 CHANGE SLOT CONTENTS
2 CHANGE DISK PARAMETERS
3 EXIT
```

輸入①告訴系統記憶咭在那裡。

WHICH SLOT？

輸入第一個記憶咭的槽，系統會要求知道該咭的種類爲128 K，64 K，32 K或16 K。

輸入咭的種類（例如128 K）。如果這張咭是在0號槽內，系統會問：

DO YOU WANT TO SAVE ROOM FOR BASIC？

輸入適當的回答。如果是N，系統會一組（BANK）一組地問，你是否要這一組？如果使用了重新安置的DOS，要記得保留含有該DOS的 64 K或128 K RAM 咭上的第2組。

對於系統中所有的記憶咭都重覆上述的步驟。接着，選第2項目以改變磁碟參數。螢幕上顯示的選擇項目如下：

①CHANGE SLOT AND DRIVE
②CHANGE DIRECTORY SIZE
③EXIT

選擇適當的項目。

（1）改變槽及驅動器以供 PSEUDO DISK用，現用值會顯示在螢幕頂部。

（2）改變目錄區段的數目，以供 PSEUDO DISK之用（未指定時的值〔DEFAULT VALUE〕＝4個區段）

在輸入了所有參數後，離開目錄，然後回到主索引（MAIN MENU），選第3項目。

檔案 PSEUDO PARAMETERS 會接著被新的參數等所更新，並顯示主索引。PSEUDO DISK此時尙未活動。

安裝PSEUDO DISK

在輸入了磁碟參數、位置，以及記憶咭的種類之後，PSEUDO DISK 就可被安裝（INSTALL）了。選第3項目。

系統會將適當的二進位程式 PSEUDO NRM 或 PSEUDO MVD 載入，這視乎使用的是否爲一個正常的或重新安置的DOS而定；並以一個空白的目錄（DIRECTORY）初設 PSEUDO DISK。該 PSEUDO DISK可以用前面所指定的槽及驅動器（未指定時爲第5號槽D1）予以取存，正如一個眞正的磁碟那樣。例如，CATALOG S5，D1，會顯示 PSEUDO DISK的目錄。

再連接PSEUDO DISK

由於 PSEUDO DISK 由特殊的驅動器及一些DOS 補充機能組成，每當系統在 PSEUDO DISK被重新啓動（RE—BOOT）之後，這些DOS補充機能會被重寫，使 PSEUDO DISK被切斷。任何在 PSEUDO DISK 上的檔案在重新啓動（只要電源仍爲開著）之後仍會在那裡，但是它們只有在驅動器常式及DOS的補充機能被重新安裝之後才能被取存。

可以用 PSEUDO DISK 程式選擇第3項目恢復這些補充功能，但是這會造成目錄被初設，及剔除任何在 PSEUDO DISK上的檔案，如選擇 PSEUDO DISK程式中主索引的第4項選擇，則可以重新安裝 DOS 補充機能而不致破壞 PSEUDO DISK目錄。

注意事項

（1）雖然在安裝及重新連接常式中已有檢查功能以防止一些 PSEUDO DISK與重新安置的DOS之間的使用記憶區的衝突，仍要小心避免無意中把程式或資料載入 PSEUDO DISK所使用的電路板的部份，而破壞了 PSEUDO DISK上的資料。（例如，將BASIC載入放在0號槽內的電路板的第1組記憶體，而該組記憶體正被 PSEUDO DISK 使用）。

（2）磁碟的初設，因爲在記憶體中的DOS的 RWTS子程式已經由補充機能來操作PSEUDO DISK，故在初設磁碟時必須小心，當在安裝有 PSEUDO DISK時初設磁碟（用 INIT 功能），補充過的DOS的一部份會被寫到磁碟上。該磁碟無法正常地啓動，因爲有了這些 DOS 補充機能，因此建議你在安裝 PSEUDO DISK 之前，用標準的DOS初設磁碟，以避免這個問題。

（3）不必每次安裝 PSEUDO DISK時，都執行它的組成部份（SET—UP PORTION），而只在其連接方式（CONFIGURATION）上有所變更時（如果變更記憶板或位置時）才執行該部份。

PSEUDO DISK的使用

只要 PSEUDO DISK 被初設及安裝（經由執行適當的 BASIC 程式或 BRUN DISK程式），就可

以當做標準磁碟機使用。它的目錄是空的，而檔案可以存放在裡面，就像一個眞正的磁碟機一樣。在擬磁碟上可以存放的檔案數目只限於它所使用的記憶體大小（32K 電路板，64K，128K，及16K咭）。

由於 PSEUDO DISK 的獨特結構（32K區段／磁軌），標準磁軌及 NIBBLE COPY 程式通常不能從 PSEUDO DISK 將資料複印。事實上，FID 必須加以修正之後才能將檔案從一個眞正的磁碟機（16個區段）轉送到 PSEUDO DISK（32個區段）之上。這種修正係由隨 RAM咭供應的磁碟上的MOD程式的執行而達成。

MOD會將FID修正，使到後者能夠在 PSEUDO DISK及重新安置的DOS裡。修正過的FID可以直接取代 FID（如同隨附於DOS 3.3 系統主程式者），不管有沒有 PSEUDO DISK 或重新安置的DOS。

只要電源維持，資料就會留在 PSEUDO DISK上，直到資料被剔除爲止。在系統被重新啓動之後，該資料仍將繼續存在。

操作方法

PSEUDO DISK 由以下三個程式構成：

（1） PSEUDO DISK：這個 APPLESOFT 程式提供在使用者與機械語言程式（PSEUDO.NRM，PSEUDO.MVD）之間的介面，該機械語言程式實際上做 DOS 的補充機能以及驅動器常式的安裝，以模仿磁碟機。本程式產生一個二進位的資料檔案 PSEUDO PARAMETERS，該檔包容有本系統的位置以及RAM電路板的種類，指定給 PSEUDO DISK 的槽，驅動器等資料。這個檔案在每次 PSEUDO DISK 被組成時（主目錄的等2個項選擇）被更新。

（2） PSEUDO、NRM：這個程式實際是PSEUDO DISK安裝以及本系統的初設（如需要的話，包括目錄）。PSEUDO DISK的驅動器常式容於檔案緩衝器與及DOS 的起始處之間，而 HIMEM 被往下調整。

（3） PSEUDO.MVD：這個程式在使用重新安置的DOS時，做PSEUDO DISK的實際安裝（由BASIC程式，PSEUDO DISK 執行）以及初設工作。PSEUDO DISK的驅動常式位在與被重新安置的 DOS的介面常式（INTERFACE ROU—TINE）的聚接下方（$BE00），而 HIMEM 被設定以保護它不致被覆寫。

CP/M 擬磁碟

CP／M PSEUDO DISK爲一個軟體套裝程式，用以使128K RAM咭可在CP／M操作系統下當做快速“RAM”磁碟機。它可用在32K，64K，128K RAM咭，與及標準的16K RAM咭。CP／M的BIOS部份已被加強一些機能，務使能將RAM咭的記憶體當做一般磁碟。因此之故，RAM咭實際上是模仿一部磁碟機，其優點是使資料傳送以及工作效率比用磁碟機更快。

本軟體的程式包括PSEUDO DISK系統，補充了CP／M，以及安裝控制常式。這些功能由以下程式執行：

PSEUDO BAS

這個程式用以組成PSEUDO DISK並包括一個可告訴系統記憶咭在何處，以及輸入磁碟參數的部份，程式將這些參數加入PSEUDO DISK安裝程式INT.COM及INIT2.COM。

PSEUDO. COM

這個機器語言程式實際上組成PSEUDO DISK磁碟機常式，其參數在組成過程中被輸入。程式由PSEUDO BAS 載入及執行。INIT. COM及INIT2.COM從PSEUDO·COM產生。

INIT.COM

這個程式執行的功能與INIT2.COM 相同，但是在安裝PSEUDO DISK磁碟機常式之後，它並不刪除目錄。

PARAM. DAT

這個檔案含有在組成過程中輸入的參數（PSEUDO DISK的磁碟機、記憶咭的位置及類別等），且由PS—EUDO·BAS 用以產生INIT. COM及INIT2.COM。

在使用PSEUDO DISK之前，必須先完成2個基本步驟：

（1） PSEUDO DISK必須被組成。

（2） PSEUDO DISK 磁碟機必須被安裝好，且系統必須被初設。

PSEUDO DISK 的組成

此步驟係有關告訴系統記憶咭在何處（那一個槽位）

，以及要使用它們的那一個部份。此外，並指定要指派給PSEUDO DISK的驅動機，以及要被載入的目錄的記憶咭數目。

要組成PSEUDO DISK時，需要以下的檔案：

PSEUDO·BAS
PSEUDO·COM
INIT·COM
INIT2·COM
PARAM·PAT

這些檔案應該被轉送到一個含有CP／M操作系統的磁碟，並被安裝在磁碟機A之內。

首先載入MBASIC並執行PSEUDO，

畫面會顯示如下的選擇項目：

```
CP/M PSEUDO DISK

1. Look at pseudo disk set-up
2. Set-up pseudo disk
3. Install pseudo disk
4. Reconnect pseudo disk
5. Exit
```

這時候，選擇第2項目。系統會接着顯示：

```
                 CP/M  PSEUDO  DISK

DRIVE C:                             DIRECTORY BLOCKS 1
SLOT            BANK        SLOT          BANK
                123456788                 12345678
0                           4
1                           5
2                           6
3                           7
--------------------------------------------------------
Y=USED          N=NOT USED       C=RESERVED FOR CP/M

1.CHANGE SLOT CONTENTS
2.CHANGE DISK PARAMETERS
3.EXIT

WHICH OPTION ?
```

現在選第一項目以輸入槽的內容。系統會接着問：

WHICH SLOT?

這時候，輸入第一個記憶咭的槽編號。

系統會接着詢問記憶咭的種類（16K，32K，64K或128K）。選擇適當的種類後，系統接着問在該記憶咭中要使用的記憶組。對每個要使用的記憶組輸入Y，對要保留的記憶組則回答NO，如記憶咭是在零號槽內系統，系統會問是否要保留空間給CP／M（即56K CP／M）。

重複上述步驟，直到所有PSEUDO DISK要使用的記憶咭都被指定爲止。接着選第2項目以輸入參數。該選擇項目會顯示：

```
1 CHANGE DRIVE
2 CHANGE NUMBER OF DIRECTORY BLOCKS
3 EXIT
```

（1）選擇第一項目是改變要被指定爲PSEUDO DISK的磁碟機（未指定時自動設定〔DEFAULT〕爲＝C：］）。

（2）選擇第二項目是改變目錄（DIRECTORY）的記憶塊數目（未指定時自動設定爲1）。

注意：每個目錄的記憶塊可以有32個目錄項目。

在所有的記憶咭都被指定且參數也輸入之後，按下3脫離程式，然後再按3以退回到主目錄（MAIN ME—NU）選擇。系統會以新參數更新PARAM DAT。在安裝PSEUDO DISK之前必須要做的最後一件事，爲產生INIT·COM及INIT2·COM檔案。這些程式便是經由補充CP／M以安裝PSEUDO DISK的程式。它們會由在組成過程第2項選擇中所輸入的參數組成。這時選第3項目以組成INIT·COM及INIT2·COM。系統會在這個組成過程中頻頻地取存磁碟及中止。

這些檔案一旦組成了，本程式將會結束，退回到CP／M系統的操作模態。

PSEUDO DISK 的安裝

在組成PSEUDO DISK之後，只須執行INIT·COM即可將它安裝。執行INIT·COM會使PSEUDO DISK的磁碟機常式被載入I／O連接記憶塊（CO—NFIGURATION BLOCK），並使CP／M BIOS補充一些機能以便使用這些磁碟機。這個初設動作的最後一個步驟將爲刪除 PSEUDO DISK的目錄。程式的INIT2·COM與INIT·COM一樣，不同的只是前者不刪除目錄。

INIT2·COM不會刪除這些檔案，而INIT COM可能會，因爲後者可能會刪除目錄。

注意：不必在每次安裝PSEUDO DISK時執行PSEUDO DISK的組成部份。只須在連接方式有所變動時，才需要組成PSEUDO DISK。(也就是說，當記憶咭的位置被改變或是參數被修正時）。事實上，在安裝PSEUDO DISK時，（在完成組成之後安裝）所必須的檔案只是INIT·COM及INIT2·COM 而已。

PSEUDO DISK 的使用方法

安妥PSEUDO DISK之後，它就可以用所指定給它的磁碟機（未指定時自動設定爲＝C：），像使用一般磁碟機。檔案可以用標準檔案複製程式如 FIP 傳送入（或出）PSEUDO DISK。

事實上，一個SUBMIT FILE 可用來：安裝P—SEUDO DISK或複製一些檔案而只需要一個指令。

例如，下述的SUBMIT FILE XFER·SUB能安裝PSEUDO DISK（DRIVE C:）並接着將檔案從磁碟機B複製到PSEUDO DISK，只需要打入SUBMIT XFER：

```
--------------------------------------------
file  XFER.SUB:
            INIT
            PIP C: = B:*.*
--------------------------------------------
```

限制及要求條件

（1）CP／M PSEUDO DISK可以使用高達256K的32K，64K，128K RAM咭或16K RAM咭。

（2）PSEUDO DISK磁碟機常式常駐於I／O連接記憶塊（CONFIGURATION BLOCK）中保留給使用者I／O磁碟機軟體用的部份。因此，這個空間在安置PSEUDO DISK之後就不能再供使用者的磁碟機使用。

PASCAL擬磁碟

PASCAL擬磁碟包含有下列的程式，能與APPLE PASCAL 1.1 合用。

PSEUDO

這是一個連接系統的程式。（也即是告訴系統記憶咭位置，這程式只須在安裝時執行，除非記憶咭被移動了。）

ATTACH：一組在需要時能自動於啓動時安裝所有的PASCAL PSEUDO—DISK（擬磁碟）的程式。如它和PASCAL PSEUDO—DISK都已被安裝，則在重新啓動時毋須動它。

FASTCOPY：一個將在任何磁碟機與PSEUDO—DISK之間做快速複製的程式。

FILEMARKER：一個在啓動時，標明（MARK）檔案名稱以被"快速複製"到PS—EUDO—DISK的程式。這在察看目錄時已經做好了。

FILEMOVER：一個以"快速複製"的速度將被標明（MARK）的檔案從任何磁碟機移到PSEUDO—DISK的程式。以作為一個系統發動程式（SYSTEM STARTUP PROGRAM）之用，來加強下列功能：

①連接到1個使用者STARTUP CODE程式。

②只載入（LOAD）所要的檔案，把只需要啓動（BOOT）的檔案留在啓動驅動器（BOOT DRIVE）。

③在重新啓動時不覆寫（OVERWRITING）檔案於PSEU—DO—DISK上。

④在做檔案轉送時，做廣泛的檢查。

SLIDESHO：一個說明程式，從PSEUDO—DISK上讀取一些"圖像"，並將它們顯示在高解像度螢幕上。

GRAPHICS：一個說明程式，說明如何將程式所產生的圖形存起來。

擬磁碟機硬件要求

要操作PASCAL擬磁碟最起碼的硬體要求為一部48K APPLE Ⅱ，至少一個磁碟機（可用1～5台磁碟機），以及1張記憶咭（32K，64K，或128K RAM咭均可），必須要是SATURN SYSTEM INC.的產品。

以下任何一種擴展咭都可以放在0號槽內使用：

① APPLE LANGUAGE CARD。

② 16K RAM CARD

③ 任何16K RAM咭（32K，16K，128K）

另外還可在其他槽內放入其他的RAM咭。在這些咭上的記憶體（多達256K）可以當做擬磁碟使用。

軟件要求

PASCAL操作系統2·1，及SATURN SYSTEMS PASCAL DISKETIE。

編目及磁碟容量

PASCAL操作系統2·1以下列次序給磁碟機編目：4，5，11，12，9，10。PSEUDO—DISK排在第10組，其餘的空間可以另安裝多達5個磁碟機。

與大部份磁碟機不同，擬磁碟記憶咭可以安裝在任何空出的槽內。最多可有 256K 供裝擬磁碟用，除去4個記憶塊供磁碟目錄，及2個記憶塊供啓動帶〔 BOOT STRAP 〕使用之外。共餘下 506 個記憶塊供使用者使用。

組成擬磁碟方法

在未組成 PASCAL擬磁碟之前，應先將 SATURN SYSTEMS PASCAL 磁碟 COPY 一隻支援磁碟。方法十分簡單，用 SYSTEM FILER便可做到。

首先，以SYSTEM將下列檔案從 SATURN SYSTEMS PASCAL磁碟轉送到啓動機磁碟。

1、PSEUDO。CODE （ 連接程式）
2、PSEUDO，DATA （ 被 SYSTEM. ATTACH 使用）
3、ATTACH. DRIVERS（同上）
4、PARAM. DATA（被 PSEUDO. CODE 使用）

然後執行 PSEUDO 即可，再從螢幕上顯示出來的目錄索引上選擇適當的項目。

FASTCOPY 使用方法

FASTCOPY 是一個很快的載入程式（ LOADER PROGRAM），被設計來在 PASCAL之下與 SATURN PSEUDO DISK 共同使用。將一整張磁碟（280個記憶塊）載入 PSEUDO DISK可以在大約 20 秒之內完成。從 PSEUDO DISK將一整張磁碟（280 個記憶塊）載入一張磁碟可以在大約 30 秒之內完成。

FASTCOPY 被設計來將迷你磁碟上的內容快速轉送到 PSEUDO DISK上。PSEUDO DISK的快速取存時間，可以利用在這些被轉送了的檔案上。

FILEMEARKER 使用方法

FILEMARKER 爲一個程式，用以顯示某特定磁碟的目錄並可藉以"標記"（ MARK） 檔案以供 FILEMOVER 程式轉送。FILEMARKER 建立一個由所選擇的檔案名稱及原來磁碟的編目名稱所組成的檔案，並將它寫到一個名爲 FILEMARKER.DATA 的前置編目（ PREFIX VOLUME）。FILEMARKER 顯示原來及目的編目的名稱與原來目錄的內容，和在目的磁碟上現有記憶塊的數目。 FILEMARKER 的目的爲建立一個檔案名稱清單，以便在啓動時，FILEMOVER能自動移動。

FILEMOVER 使用方法

FILEMOVER 爲一個公用程式，設計來將存貯在某張磁碟上的一些檔案的名稱的資料檔案(FILEMARKER. DATA)讀入， 並將這些檔案從該磁碟轉送到 PSEUDO —DISK。 它被設計爲一種發動程式（ START—UP），其功能是預先載入 PSEUDO—DISK 與一些當系統開動時使用者就需要的檔案。另有一種功能是FILEMOVER被執行完之後執行一個使用者的發動程式。（USE START UP PROGRAM)。

64/128 RAM 咭的結構

RAM 咭設址方法

64K RAM咭含有64K BYTE 組（如爲 128 K RAM 則有 128 K BYTE組），並分成16K記憶組16K BYTE BANK 。這些記憶組可各自獨立選用。

所有的16K記憶組與 APPLE 主機板 ROM佔用同樣的記憶位置，即$D000～$FFFF。就這點而言，它們與ROM共用相同的記憶空間。在任何時候，只能從主機板上的 ROM或是16K記憶組中取存其中一組，決不可兩者同時取存。因爲ROM只有$D000～$FFFF的12K空間，而RAM咭的每個記憶組有16K，故這些記憶組只有12K可以取存。原因是從$C000～$CFFF的4K記憶體被 APPLE Ⅱ 用來做 I／O，而不能與64K或 128 K咭上的 RAM 共用。

爲取存 RAM咭上每個記憶組的16K記憶組，每張 RAM咭的低位 4K都被使用兩次。每張 RAM 咭被分成二個佔住$D000～$DFFF的4K記憶組，以及一個佔住$E000～$FFFF 的8K記憶組。在一特定的16K記憶組上，使用者可以選擇二組 4K記憶組中，任何佔用$D000～$DFFF空間的一組，使用者可以用12K的記憶段（ SEGMENTS）取存所有在64K電路板(在 128 K電路板）上的64K。要定義某特定段時，必須指定二個東西：

（1） 要取存的16K記憶組。

（2） 在該16K記憶組內要使用的4K。

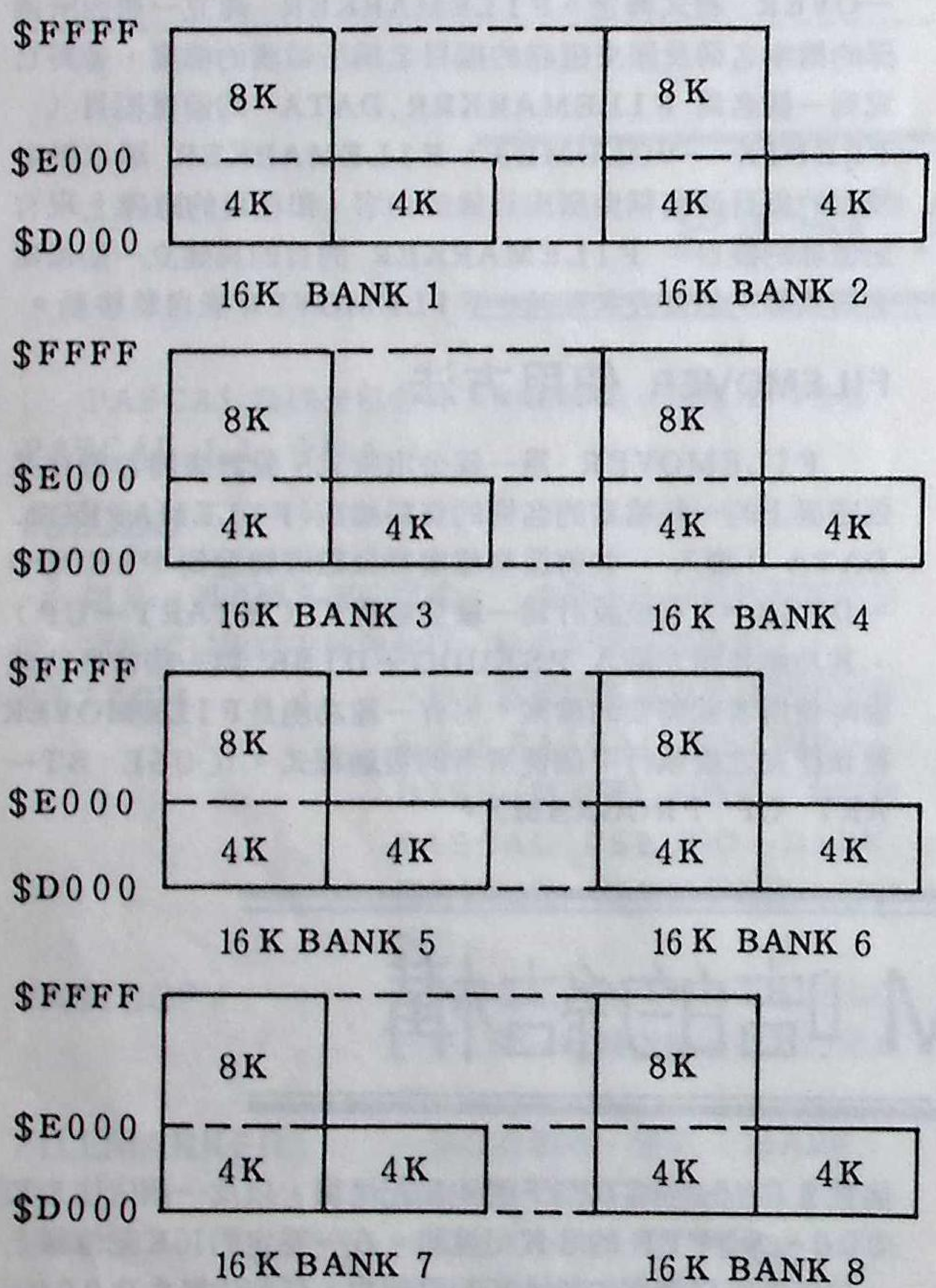
以下的規則已被用來指定64K RAM 咭上的各部分：四個16K記憶組分別被指定爲16K BANK 1，16K BANK 2，16K BANK 3，16K BANK 4。 在每個16K記憶組內，有兩個4K記憶組再被指定爲4K記憶組A，4K記憶組B。要將第一個16K BANK 1 內的4K記憶組與第2、3、4個16K BANK 內者區別起見，在4K記憶組的名稱之前加上1、2、3、4，以表示所屬的16K記憶組。因此，4K記憶組1A，表示在16K BANK 1之內的第一個4

K記憶組。

128 K RAM 咭的指定方式與64K RAM 咭相同，差別只是前者含有八個16K記憶組而非四個。這記憶組被指定爲16K BANK 1 到16K BANK 8。

下面的記憶區圖說明在128 K RAM咭記憶組的結構，在64K RAM 咭只說明前面四個16K記憶組。

記憶區位置



RAM 咭的控制

64K及128 K RAM咭用16個 $C0N0～$C0NF的位置控制，其中N＝8＋ RAM咭的槽編號。因此對於0號槽的操作，其控制地址爲$ C080～$C08F。這些位置在APPLE 的週邊輸出／入空間內。64K及128 K電路板用這些地址控制下列功能：

（1）選擇RAM咭或 APPLE 主電路板上的ROM。

（2）選擇所要的16K記憶組以取存。

（3）決定在每個16K記憶組內要使用那個4K記憶組（A或B）。

（4）將 RAM開用以寫入或防止寫入。

要將64K或128 K RAM電路板設定在某一特定模態（STATE）時，只須存記憶體中相當於所要的模態位置。這可經由讀或寫該位置即能達成。BASIC中的 POKE 或 PEEK 可完成這種取存的工作。機械語言的載入或存放也可有相同的作用。控制位置內的四個低位位元選擇64K及128K電路板的操作模式，並決定要取存那一個16K記憶組。

A、當地址行2是在低（A＝0）模態時，就可進行選擇。在這種情況下，地址行0，1及3決定模態，其設定方法如下：

（1）地址行0決定電路板是否被開用可開始寫入，或防止寫入（A0＝0爲防止寫入；A0＝1爲可開始寫入）。

（2）地址行0及1用以決定所取存的究竟是主電路板ROMS 或在 SATURN 電路板上的RAM。

（3）地址行3決定要使用所選的16K記憶組中的那一個4K記憶組。

B、反過來說，當地址行2爲高位（A2＝1）時，可開始做16K記憶組的選擇。地址行0，1及3決定選擇那一個16K記憶組。

附表A爲各種可能的模式結合：

CONTROL ADDRESS	FUNCTION
$CON0	4K BANK A;RAM READ ;WRITE PROTECT
$CON1	4K BANK A,ROM READ ,WRITE ENABLED
$CON2	4K BANK A,ROM READ ,WRITE PROTECT
$CON3	4K BANK A,RAM READ ,WRITE ENABLED
$CON8	4K BANK B,RAM READ ,WRITE PROTECT
$CON9	4K BANK B,ROM READ ,WRITE ENABLED
$CONA	4K BANK B,ROM READ ,WRITE PROTECT
$CONB	4K BANK B,RAM READ ,WRITE ENABLED
$CON4	SELECT 16K BANK 1
$CON5	SELECT 16K BANK 2
$CON6	SELECT 16K BANK 3
$CON7	SELECT 16K BANK 4
$CONC	SELECT 16K BANK 5
$COND	SELECT 16K BANK 6
$CONE	SELECT 16K BANK 7
$CONF	SELECT 16K BANK 8

附表A

適當地使用模式結合（附表A），就可以取存 RAM咭的任何位元組。事實上，在一些模式下（C0N1，C0N9），主機板的 ROM在從 $D000～$FFFF 讀取時被取存，而RAM 咭上當執行寫入時被寫入。這樣使得者可以在執行 ROM中的 BASIC 時，同時寫入 RAM（BASIC佔用同樣的地址空間）。這是可能的，因爲當程或可能在寫入RAM（用 POKES）時，BASIC 正被從 ROM讀入。

RAM 咭用途

(1)重新安放 DOS

重新安放過的DOS 佔住第二個16K記憶組及4K記憶組中的2A，使用全部12K的記憶區 $D000~$FFFF，以供 DOS，監督常式（MONITOR ROUTINES），以及相關的檔案緩衝器。這個記憶組係以控制地址 $C0N3及記憶選擇地址$C0N5開用的。4K記憶組2B可被重新安放過的 DOS 使用。

(2)任何一種BASIC

當任何一種 BASIC 在 RAM咭上時，它佔住16K BANK 1 及4K記憶組1A，並且以控制地址$C0ND及記憶組選擇 $C0N4開用供讀取。這個 BASIC 未使用4K記憶組1B。

(3)LED指示燈

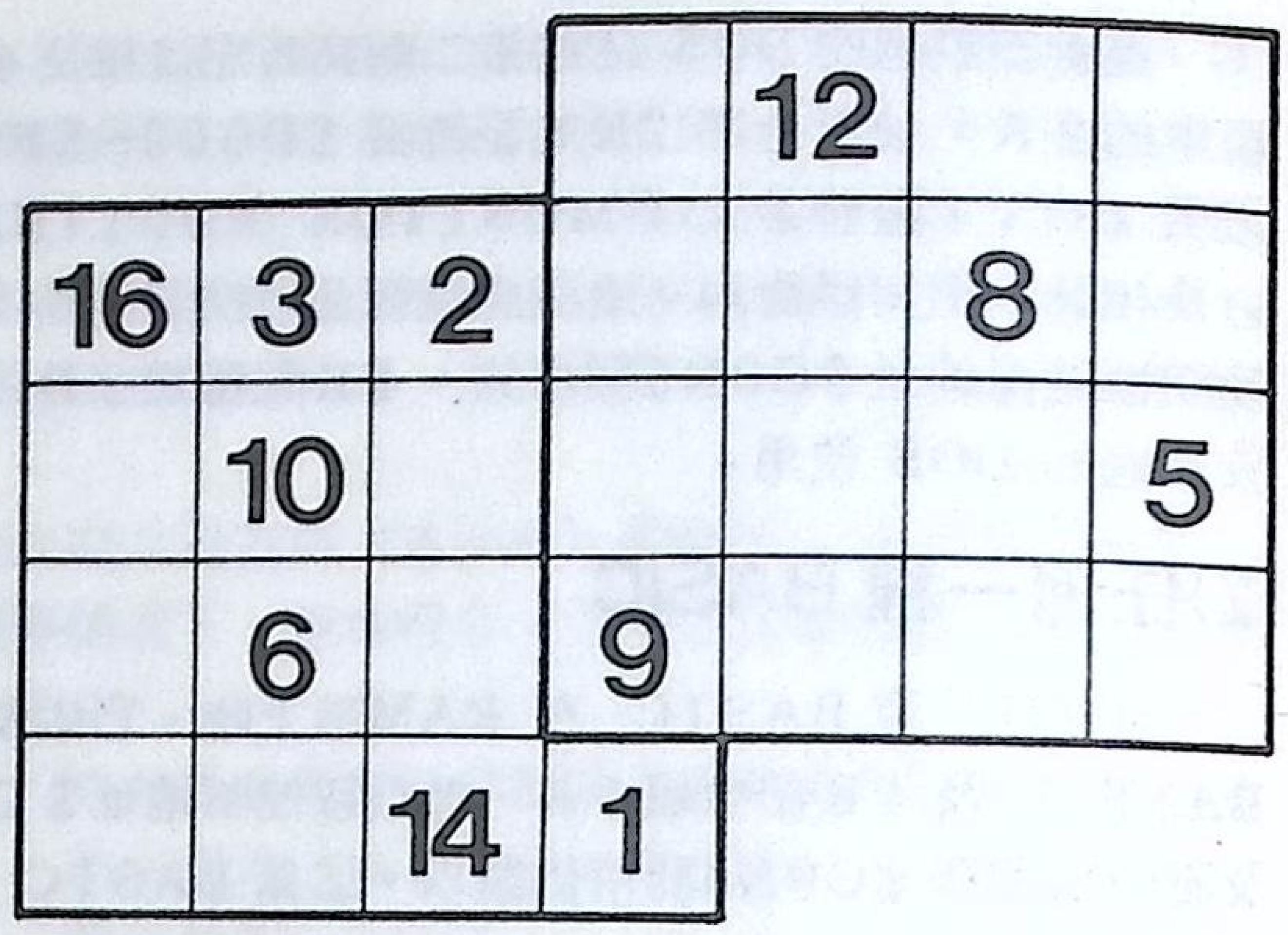
64K及128 RAM 咭有四枚 LED 指示電路板的狀態。其中三個用的指示那一組記憶組被選用，而第四個用以指示電路板是否被開用可供讀取（RAM READ模態）。

LED 所顯示的意義如下：（當電路板爲面朝上，且LED 在電路板的頂上時，從左到右順序）A3，A1，A0，RAM READ 。當電路板被開用可讀取時（$C0N0，$C0N3，$C0N8，$C0NB），RAM READ LED爲〝ON〞

LED 有16種可能的模態。

BANK,	STATE	A3	A1	AO	RAM READ
16K BANK 1	ROM READ	OFF	OFF	OFF	OFF
16K BANK 2	ROM READ	OFF	OFF	ON	OFF
16K BANK 3	ROM READ	OFF	ON	OFF	OFF
16K BANK 4	ROM READ	OFF	ON	ON	OFF
16K BANK 5	ROM READ	ON	OFF	OFF	OFF
16K BANK 6	ROM READ	ON	OFF	ON	OFF
16K BANK 7	ROM READ	ON	ON	OFF	OFF
16K BANK 8	ROM READ	ON	ON	ON	OFF
16K BANK 1	RAM READ	OFF	OFF	OFF	ON
16K BANK 2	RAM READ	OFF	OFF	ON	ON
16K BANK 3	RAM READ	OFF	ON	OFF	ON
16K BANK 4	RAM READ	OFF	ON	ON	ON
16K BANK 5	RAM READ	ON	OFF	OFF	ON
16K BANK 6	RAM READ	ON	OFF	ON	ON
16K BANK 7	RAM READ	ON	ON	OFF	ON
16K BANK 8	RAM READ	ON	ON	ON	ON

神奇數字方塊



相信大部份的讀者都曾經嘗試過神奇方塊這玩意兒，它的起源最早始於中東和印度的數學。

它需要的是把一連串的整數，由1開始，放在一個由小格子組織成的方塊內。放這些整數的要求是，無論橫豎或對角綫上，把小格中的數目加起來，總和永遠是一個常數。

這個常數可由下列的方程式求得：—

$(N^3+N) / 12$ ；

N是在任何一邊的小格子的數目。

通常引起注意的神奇方塊，多是那些有4×4個或5×5個小格子的。（3×3個小格子的神奇方塊，只有一個答案）4×4的神奇方塊的那個常數是 34，而5×5的是65。

4×4的神奇方塊可以有數百個的變化，每一單位性的增值都戲劇化地增加。究竟這些方塊有甚麼地方令我們激發起那麼大的興趣呢？那就是當我們將單／雙或全部數目，順序地連起來，我們便會得到一些有趣的幾何模式。

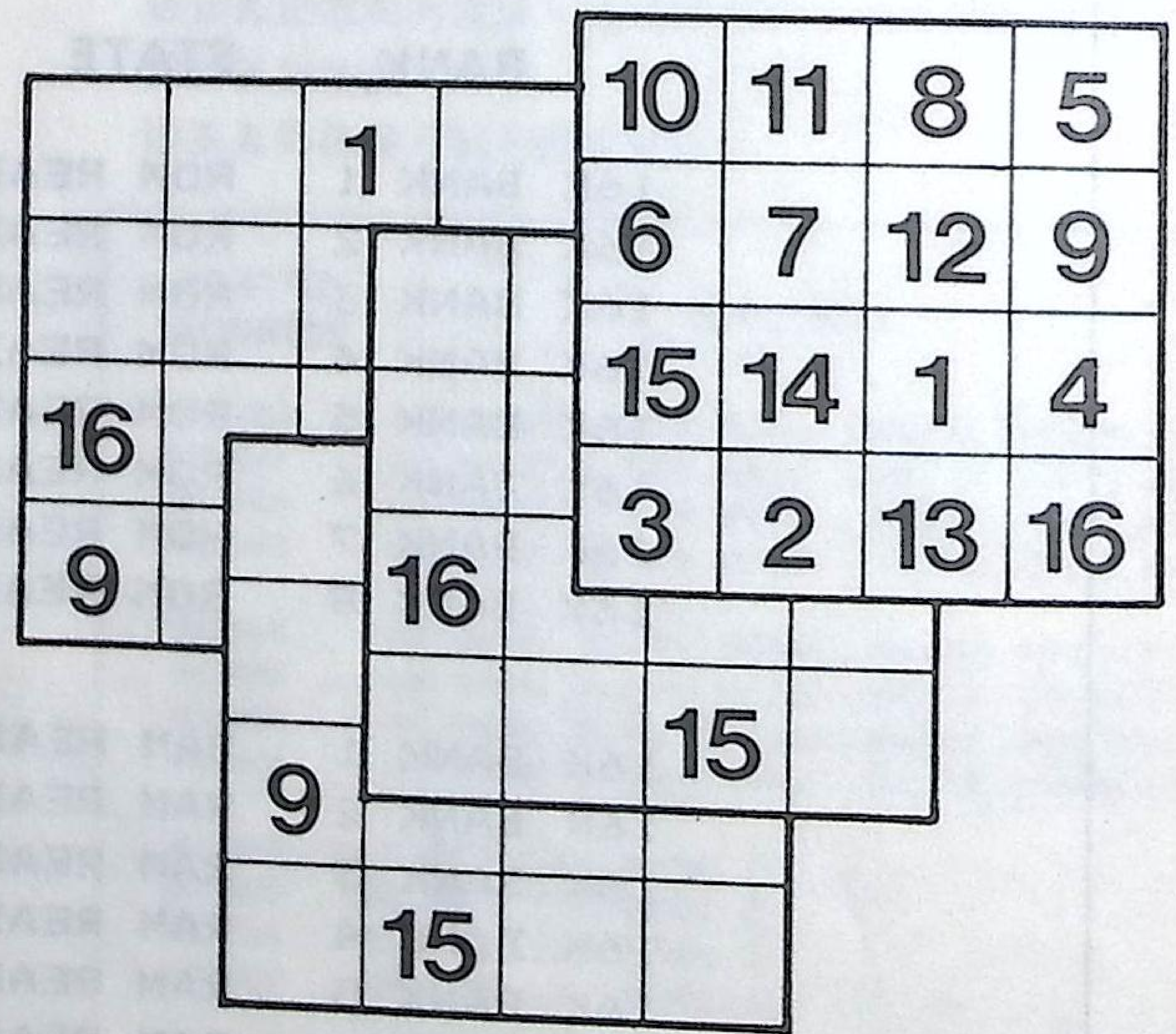
所有這些都普遍地爲大家所知，同時看起來，把這些模式的產生，去用高解像螢幕顯示，說起來是一個有趣的題材啊！

程式的本身把一些經過選擇過的神奇方塊，作爲資料地讀入，跟着把它們放去文字畫面。

轉去用高解像螢幕，我們便可給出這些由三個不同的方塊演譯出來的幾何模式了。

這個程式不太難，有興趣的讀者可以自己輸入神奇方塊的資料，產生更神奇的方塊呢！

本程式在程式磁碟中的 FILENAME 是 MAGIC 。□

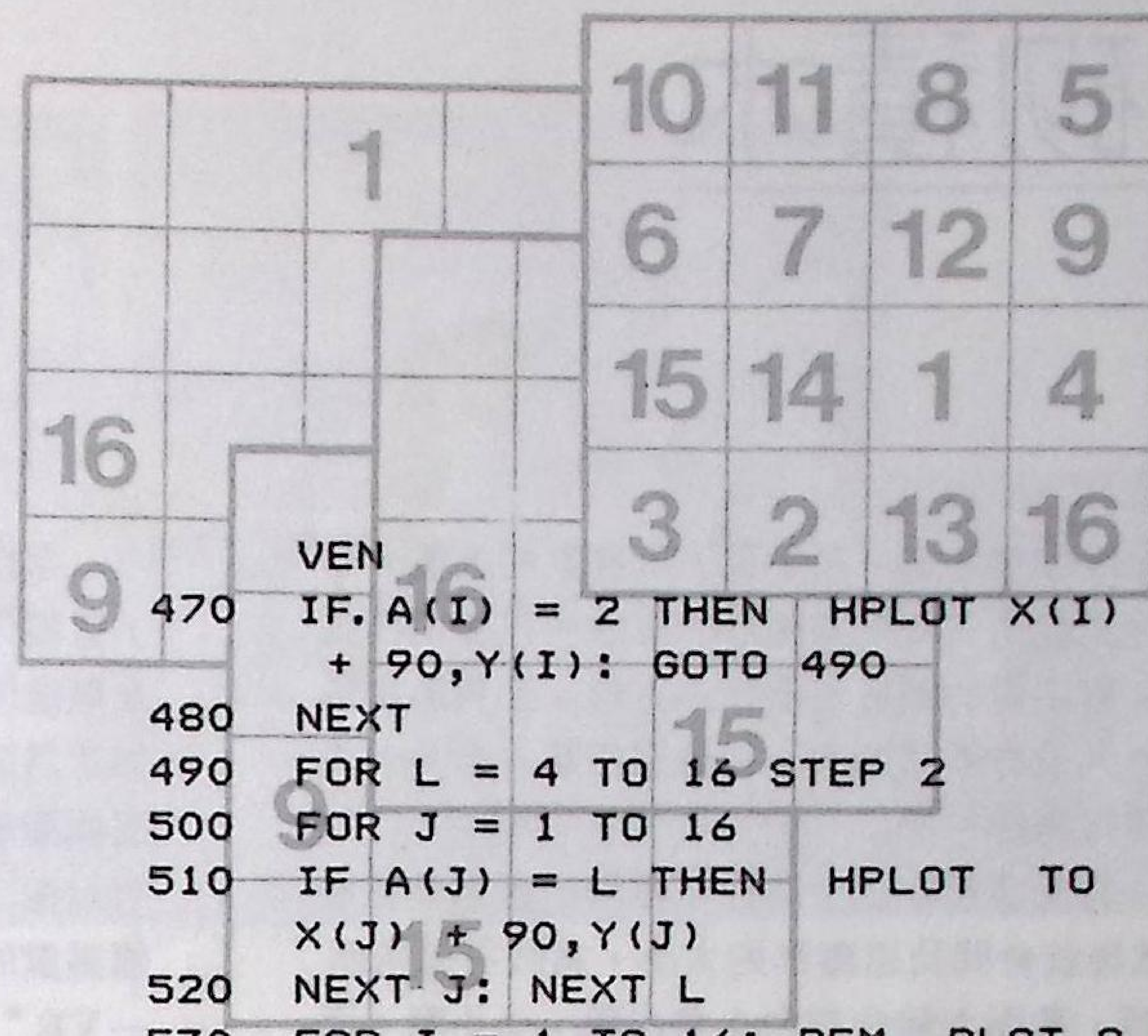


MAGIC

```
15  HOME
20  DIM X(16),Y(16),A(16)
30 COUNT = 0
40  FOR I = 1 TO 16: READ X(I): NEXT

50  FOR I = 1 TO 16: READ Y(I): NEXT

60  FOR I = 1 TO 16: READ A(I): NEXT
    : REM  SQUARES DATA
70 HT = 1
80  HOME : TEXT
90  VTAB (1): HTAB (11): GOSUB 65
    0
100  HTAB (11): PRINT  CHR$ (255)
    ;: PRINT "MAGIC SQUARE NO.";
    COUNT + 1;: PRINT  CHR$ (255
    )
110  HTAB (11): GOSUB 650
120  VTAB (4): HTAB (12)
130  FOR S = 1 TO 9 STEP 2
140  VTAB (S + 4): HTAB (14)
150  GOSUB 640
```




```
160  NEXT
170  VTAB (6)
180  FOR I = 1 TO 16: REM  SQUARE
      TO TEXT SCREEN
190  HTAB (HT + 13): PRINT  CHR$
     (255);
200  HTAB (26): PRINT  CHR$ (255)
     ;
210 A$ =  STR$ (A(I)):A =  LEN (A
     $)
220  IF A = 2 THEN  HTAB (HT + 14
     ): PRINT A(I);: GOTO 240
230  IF A = 1 THEN  HTAB (HT + 15
     ): PRINT A(I);
240 HT = HT + 3
250  IF I = 4 OR I = 8 OR I = 12 THEN
     HT = 1: PRINT : PRINT
260  NEXT
270  PRINT : PRINT : PRINT
     "IN A 4X4 CELL MAGIC SQUARE
     THE SUM OF"
280  PRINT : PRINT "ANY SIDE,HORI
     ZONTAL,OR VERTICAL"
290  PRINT : PRINT "WILL BE TOTAL
      34."
300  HTAB (9): VTAB (24): INVERSE
     : PRINT "ANY KEY TO CONTINUE
     ";: NORMAL
310  VTAB (24): HTAB (31): GET Q$

320  HTAB (1): VTAB (21): PRINT "
       PLOT ALL    PLOT EVEN
      PLOT ODD"
330  PRINT "    -NUMBERS------NUMB
     ERS------NUMBERS-    "
340  HGR : HCOLOR= 3
350  HPLOT 10,40 TO 90,40 TO 90,1
     20 TO 10,120 TO 10,40
360  HPLOT 100,40 TO 180,40 TO 18
     0,120 TO 100,120 TO 100,40
370  HPLOT 190,40 TO 270,40 TO 27
     0,120 TO 190,120 TO 190,40
380  FOR I = 1 TO 16: HPLOT X(I),
     Y(I): HPLOT X(I) + 90,Y(I): HPLOT
      X(I) + 180,Y(I): NEXT
390  FOR I = 1 TO 16: REM  PLOT A
     LL
400  IF A(I) = 1 THEN  HPLOT X(I)
     ,Y(I): GOTO 420
410  NEXT
420  FOR L = 2 TO 16
430  FOR J = 1 TO 16
440  IF A(J) = L THEN  HPLOT  TO
     X(J),Y(J)
450  NEXT J: NEXT L
460  FOR I = 1 TO 16: REM  PLOT E
     VEN
470  IF A(I) = 2 THEN  HPLOT X(I)
      + 90,Y(I): GOTO 490
480  NEXT
490  FOR L = 4 TO 16 STEP 2
500  FOR J = 1 TO 16
510  IF A(J) = L THEN  HPLOT  TO
     X(J) + 90,Y(J)
520  NEXT J: NEXT L
530  FOR I = 1 TO 16: REM  PLOT O
     DD
540  IF A(I) = 1 THEN  HPLOT X(I)
      + 180,Y(I): GOTO 560
550  NEXT
560  FOR L = 3 TO 16 STEP 2
570  FOR J = 1 TO 16
580  IF A(J) = L THEN  HPLOT  TO
     X(J) + 180,Y(J)
590  NEXT J: NEXT L
600  VTAB (24): HTAB (30): GET Q$

610 COUNT = COUNT + 1
620  IF COUNT = 8 THEN  END
630  GOTO 60
640  FOR I = 1 TO 13: PRINT  CHR$
     (255);: NEXT : RETURN
650  FOR I = 1 TO 19: PRINT  CHR$
     (255);: NEXT : PRINT : RETURN

660  END
670  DATA   20,40,60,80,20,40,60,
     8          0,20,40,60,80,2
     0,40,60,80,50              ,50
     ,50,50,70,70,70,70,90,90,
               90,90,110,110,110,1
     10 : REM                X,Y C
     OORDINATES FOR HI-RES
680  DATA    10,11,8,5,6,7,12,9,1
     5,14,1,4,3,2,13,16: REM  DAT
     A FOR SELECTION OF MAGIC SQU
     ARES
690  DATA 16,3,2,13,5,10,11,8,9,
     6,7,12,4,15,14,1
700  DATA 7,12,1,14,2,13,8,11,16
     ,3,10,5,9,6,15,4
710  DATA  1,8,13,12,14,11,2,7,4,
     5,16,9,15,10,3,6
720  DATA  16,2,3,13,5,11,10,8,9,
     7,6,12,4,14,15,1
730  DATA  7,12,9,6,14,1,4,15,2,1
     3,16,3,11,8,5,10
740  DATA  15,10,8,1,6,3,13,12,9,
     16,2,7,4,5,11,14
750  DATA  3,13,12,6,16,2,7,9,5,1
     1,14,4,10,8,1,15
760  REM
```

砌積木

堆砌積木是一項小孩子所喜歡的遊戲，而這程式就是一個堆砌圖形的遊戲。在這個程式中，有不同的圖形。例如正方形、三角形和長方形。也有不同的大小，分爲四類，合共十六個選擇給你。

首先這個程式會問你所雖要的圖形是甚麼？然後就會問及這圖形的大小，當回答這兩個問題，畫面上就會在左上角出現，一小點。而你可以控制這小點在畫面上移動。控制鍵如下：

"U"控制它向上移。

"D"控制它向下移。

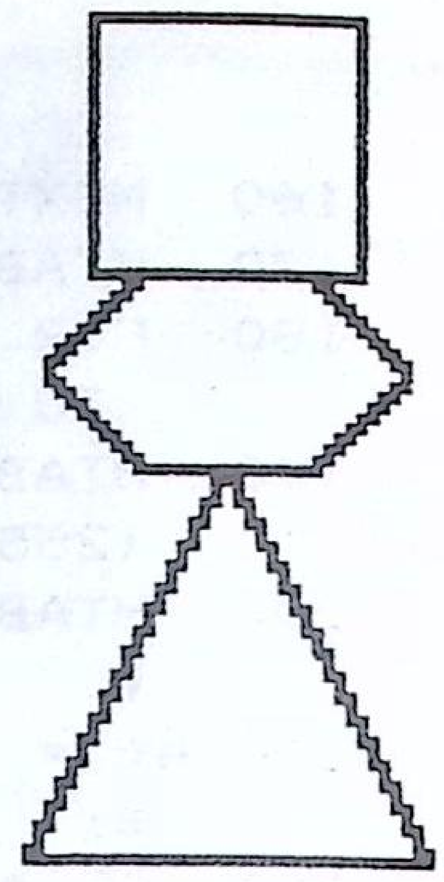
"→"控制它向右移。

"←"控制它向左移。

當你認爲位置正確時，就要 RETURN，這樣你所需要的圖形就會在畫面上出現。如此類推你就可以砌出你所喜歡的圖形來。如果你認爲這是一件傑作，你還可以" SAVE"這個圖形落磁碟上。但如果不滿意這個圖形，可以按"C"來抹去現有的圖形而從新再畫一幅滿意的，你還可以" LOAD" 以前" SAVE"起的圖形來欣賞的。

希望你的孩子能變爲一個偉大的砌積木家！

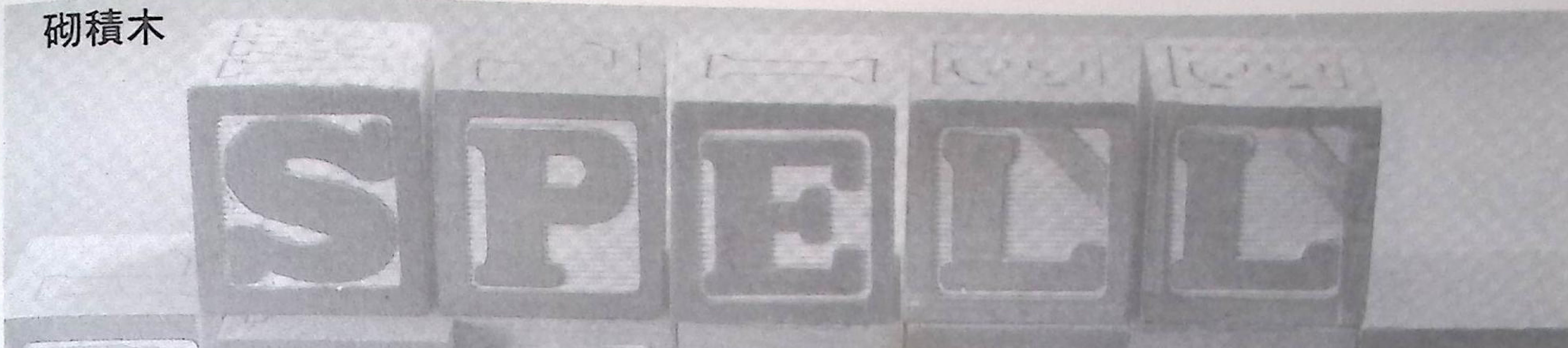
本程式於磁碟上的 FILENAME 爲 GREAT BUILDER GREAT BUILDER 程式。□





```
1   REM  *******************
2   REM
3   REM       GREET BUILDER
4   REM  **   COMPU-AGE 1984 **
5   REM
6   REM
8   DIM F$(50): DIM S$(50): DIM GX
     $(50): DIM GY$(50): DIM R$(5
     0)
10  HIMEM: 8192
20  FOR OP = 1 TO 4: READ OP$(OP)
     : NEXT
30  FOR CH = 1 TO 4: READ CH$(CH)
     : NEXT
40  GOSUB 4800
50  HOME
70 SP = 6: REM  LOCATOR SPEED
80  HGR : HCOLOR= 3
90 N = 0
95  POKE  - 16368,0
99  REM     BUIDING OPTIONS
100  IF N = 50 THEN  HOME : VTAB
     22: FLASH : PRINT "50 BLOCKS
      HAVE BEEN USED - PLEASE CLE
     AR ": NORMAL : FOR X = 1 TO
     2000: NEXT : HOME
102  FOR OP = 1 TO 4: VTAB 21: HTAB
     OP * 10 - 8: PRINT D$(OP): NEXT
105 OP = 1
110  INVERSE : VTAB 21: HTAB 2: PRINT
     D$(1): NORMAL
120  IF FLAG = 0 THEN  VTAB 23: PRINT
     "C = CLR * S = SVE * L = LD
     * ESC = END "
130 X =  PEEK ( - 16384): IF X <
     127 THEN  GOTO 130
135  IF X = 155 THEN  TEXT : HOME
     : END
138  IF X = 195 THEN  GOTO 80
140  IF X = 141 AND FLAG = 0 THEN
     N = N + 1:F$(N) =  STR$ (OP)
     : ON OP GOSUB 1000,1300,1600
     ,1850: GOTO 100
142  IF X = 141 AND FLAG = 1 THEN
     S$(N) =  STR$ (OP): RETURN
144  IF X = 211 THEN  GOSUB 3000:
     GOTO 100
148  IF X = 204 THEN  GOSUB 3500:
     GOTO 100
150 OP = OP + 1
155  IF OP = 5 THEN  POKE  - 1636
     8,0: GOTO 102
160  NORMAL : VTAB 21: HTAB (OP -
     1) * 10 - 8: PRINT D$(OP - 1
     )
170  INVERSE : VTAB 21: HTAB OP *
     10 - 8: PRINT D$(OP): NORMAL
180  POKE  - 16368,0: GOTO 130
199  REM     LOCATE OBJECT
200  HOME : VTAB 22: PRINT "U = "
     ;: INVERSE : PRINT "UP";: NORMAL
     : PRINT "  D= ";: INVERSE : PRINT
     "DOWN";: NORMAL : PRINT "
     <- = ";: INVERSE : PRINT "LE
     FT";: NORMAL : PRINT "  -> =
     ";: INVERSE : PRINT "RIGHT"
     : NORMAL
202 GX = 0:GY = 0
205  POKE  - 16368,0
207  FOR LL = 1 TO 2 STEP 0
210 X =  PEEK ( - 16384): HCOLOR=
     0: HPLOT GX,GY: HCOLOR= 3: HPLOT
     GX,GY: IF X < 127 THEN  GOTO
     210
220  POKE  - 16368,0
225  IF X = 141 THEN  RETURN
```

砌積木

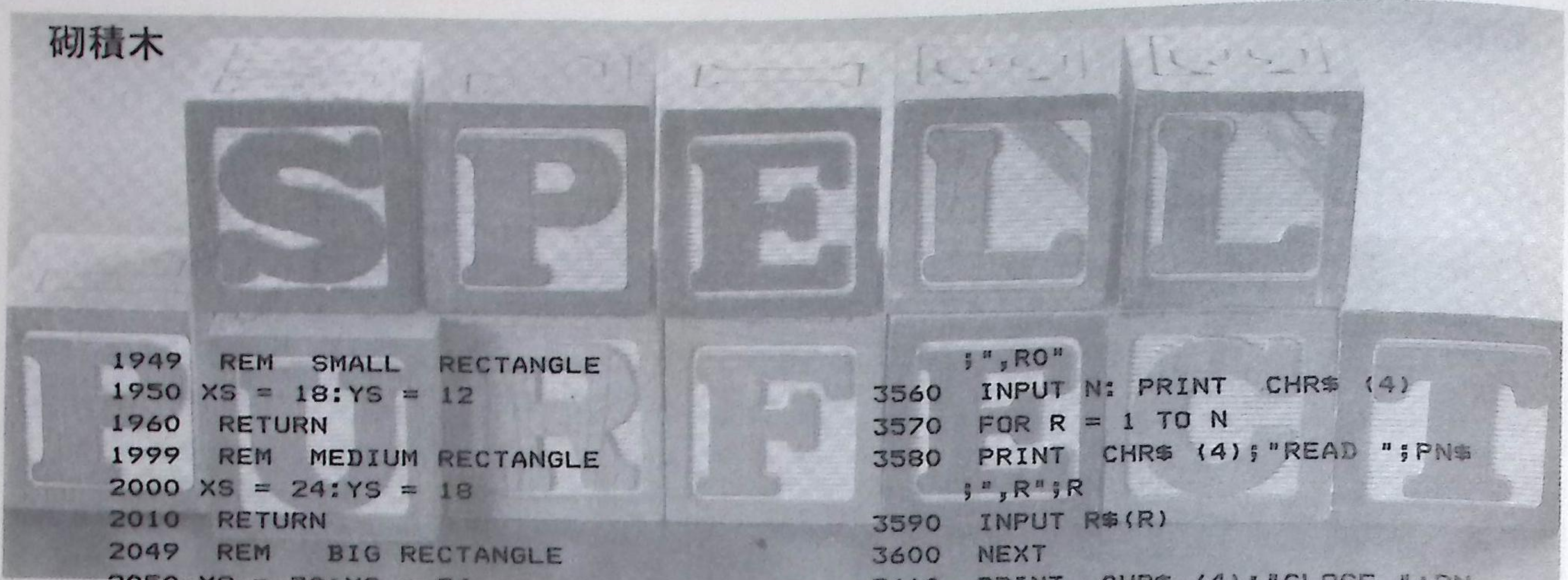



```
230  HCOLOR= 0: HPLOT GX,GY: HCOLOR=
     3
240  IF X = 149 THEN GX = GX + SP
     : REM  RIGHT
250  IF X = 136 THEN GX = GX - SP
     : REM  LEFT
260  IF X = 213 THEN GY = GY - SP
     : REM  UP
270  IF X = 196 THEN GY = GY + SP
     : REM  DOWN
273  IF GX > 279 THEN GX = 279
274  IF GY > 159 THEN GY = 159
275  IF GX < 0 THEN GX = 0
276  IF GY < 0 THEN GY = 0
280  NEXT
299  REM    KEEP ON SCREEN
300  IF GX - XS < 0 THEN GX = GX +
     1: GOTO 300
310  IF GX + XS > 279 THEN GX = G
     X - 1: GOTO 310
320  IF GY - YS < 0 THEN GY = GY +
     1: GOTO 320
340 GX$(N) =  STR$ (GX):GY$(N) =
      STR$ (GY)
350  RETURN
399  REM     SET FLAG/GET  SUB
400  POKE  - 16368,0: GOSUB 4850:
      GOSUB 100
410  POKE  - 16368,0
420  GOSUB 200
430  HOME : RETURN
498  REM      PLOT
499  REM  **SQR & RECTANGLE **
500  HPLOT GX,GY TO GX - XS,GY TO
     GX - XS,GY - YS TO GX + XS,G
     Y - YS TO GX + XS,GY TO GX,G
     Y: RETURN
509  REM  **HEXAGON**
510  HPLOT GX,GY TO GX - XS / 2,G
     Y TO GX - XS,GY - YS / 2 TO
     GX - XS / 2,GY - YS TO GX +
     XS / 2,GY - YS TO GX + XS,GY
      - YS / 2 TO GX + XS / 2,GY TO
     GX,GY
511  RETURN
519  REM  **TRIANGLE**
520  HPLOT GX,GY TO GX - XS,GY TO
     GX,GY - YS TO GX + XS,GY TO
     GX,GY: RETURN
999  REM      SQUARE
1000  GOSUB 400
1010  ON OP GOSUB 1050,1095,1140,
     1190
1012  GOSUB 300
1014  GOSUB 500
1020  GOSUB 4800
1030  RETURN
1049  REM  TINY SQR & RECTANGLE

1050 XS = 6:YS = 12
1080  RETURN
1094  REM  SMALL SQR & RECTANGLE

1095 XS = 12:YS = 24
1100  RETURN
1139  REM  MEDIUM SQR & RECTANGL
     E
1140 XS = 18:YS = 36
1150  RETURN
1189  REM    BIG SQR & RECTANGLE

1190 XS = 24:YS = 48
1200  RETURN
1299  REM  HEXAGON
1300  GOSUB 400
1310  ON OP GOSUB 1400,1450,1500,
     1550
1320  GOSUB 300
1330  GOSUB 510
1340  GOSUB 4800
1350  RETURN
1399  REM  TINY HEX
1400 XS = 10:YS = 12
1410  RETURN
1449  REM  SMALL HEX
1450 XS = 16:YS = 18
1460  RETURN
1499  REM  MEDIUM HEX
1500 XS = 22:YS = 24
1510  RETURN
1549  REM  BIG HEX
1550 XS = 28:YS = 30
1560  RETURN
1599  REM  TRIANGLE
1600  GOSUB 400
1610  ON OP GOSUB 1050,1095,1140,
     1190
1612  GOSUB 300
1614  GOSUB 520
1620  GOSUB 4800
1630  RETURN
1849  REM  RECTANGLE
1850  GOSUB 400
1860  ON OP GOSUB 1900,1950,2000,
     2050
1862  GOSUB 300
1864  GOSUB 500
1870  GOSUB 4800
1880  RETURN
1899  REM  TINY RECTANGLE
1900 XS = 12:YS = 6
1910  RETURN
```




```
1949  REM  SMALL  RECTANGLE
1950 XS = 18:YS = 12
1960  RETURN
1999  REM  MEDIUM RECTANGLE
2000 XS = 24:YS = 18
2010  RETURN
2049  REM   BIG RECTANGLE
2050 XS = 30:YS = 24
2060  RETURN
2999  REM  SAVE PICTURE
3000  FOR R = 1 TO N
3020  IF  LEN (GX$(R)) < 3 THEN G
    X$(R) = "0" + GX$(R): GOTO 3
    020
3040  IF  LEN (GY$(R)) < 3 THEN G
    Y$(R) = "0" + GY$(R): GOTO 3
    040
3050 R$(R) = F$(R) + S$(R) + GX$(
    R) + GY$(R)
3060  NEXT
3070  POKE  - 16368,0
3100  HOME : VTAB 22: INPUT "PICT
    URE NAME ";PN$
3110  PRINT  CHR$ (4);"OPEN ";PN$
    : PRINT  CHR$ (4);"DELETE ";
    PN$
3120  PRINT  CHR$ (4);"OPEN ";PN$
    ;",L9"
3130  PRINT  CHR$ (4);"WRITE ";PN
    $;",R0"
3140  PRINT N: PRINT  CHR$ (4)
3150  FOR R = 1 TO N
3160  PRINT  CHR$ (4);"WRITE ";PN
    $;",R";R
3170  PRINT R$(R)
3180  NEXT
3190  PRINT  CHR$ (4);"CLOSE ";PN
    $
3200  HOME : RETURN
3399  REM  CATALOG FOR LOAD
3400  TEXT : HOME
3410  PRINT  CHR$ (4);"CATALOG "
3420  PRINT : INPUT "<RETURN> ";R
    $
3430  RETURN
3499  REM   LOAD PICTURE
3500  HGR : HCOLOR= 3
3510  ONERR  GOTO 6000
3515  POKE  - 16368,0
3520  HOME : VTAB 22: INPUT "PICT
    URE NAME ";PN$
3530  IF PN$ = "?" THEN  GOSUB 34
    00: GOTO 3500
3540  PRINT  CHR$ (4);"OPEN ";PN$
    ;",L9"
3550  PRINT  CHR$ (4);"READ ";PN$
    ;",R0"
3560  INPUT N: PRINT  CHR$ (4)
3570  FOR R = 1 TO N
3580  PRINT  CHR$ (4);"READ ";PN$
    ;",R";R
3590  INPUT R$(R)
3600  NEXT
3610  PRINT  CHR$ (4);"CLOSE ";PN
    $
3620  FOR R = 1 TO N
3630 FS =  VAL ( LEFT$ (R$(R),1))
3640 SS =  VAL ( MID$ (R$(R),2,1)
    )
3650 GX =  VAL ( MID$ (R$(R),3,3)
    )
3660 GY =  VAL ( RIGHT$ (R$(R),3)
    )
3670  ON FS GOSUB 3800,3850,3900,
    3950
3680  NEXT
3690  HOME : RETURN
3800  ON SS GOSUB 1050,1095,1140,
    1190
3810  GOSUB 500
3820  RETURN
3850  ON SS GOSUB 1400,1450,1500,
    1550
3860  GOSUB 510
3870  RETURN
3900  ON SS GOSUB 1050,1095,1140,
    1190
3910  GOSUB 520
3920  RETURN
3950  ON SS GOSUB 1900,1950,2000,
    2050
3960  GOSUB 500
3970  RETURN
4799  REM   PREPARE FOR DISPLAY
4800  FOR DI = 1 TO 4:D$(DI) = OP
    $(DI): NEXT
4810 FLAG = 0: HOME
4820  RETURN
4850  FOR DI = 1 TO 4:D$(DI) = CH
    $(DI): NEXT
4860 FLAG = 1: HOME
4870  RETURN
4999  REM  DATA
5000  DATA  "SQUARE","HEXAGON","T
    RIANGLE","RECTANGLE"
5010  DATA  "TINY","SMALL","MEDIU
    M","BIG"
5999  REM  ERROR
6000  HOME : VTAB 22: PRINT "THER
    E HAS BEEN AN ERROR ": FOR X
     = 1 TO 6000: NEXT : RUN ■
```

迷宮搜秘

迷宮搜秘是一個頗爲有水準的遊戲。玩者必須要控制自己前進的方向，避免碰到迷宮的牆壁或走入死胡同。

前速的速度共有十個程度可供選擇。迷宮是隨機產生。控制前進方法的按鍵如下：

A——上

Z——下

→——右

←——左

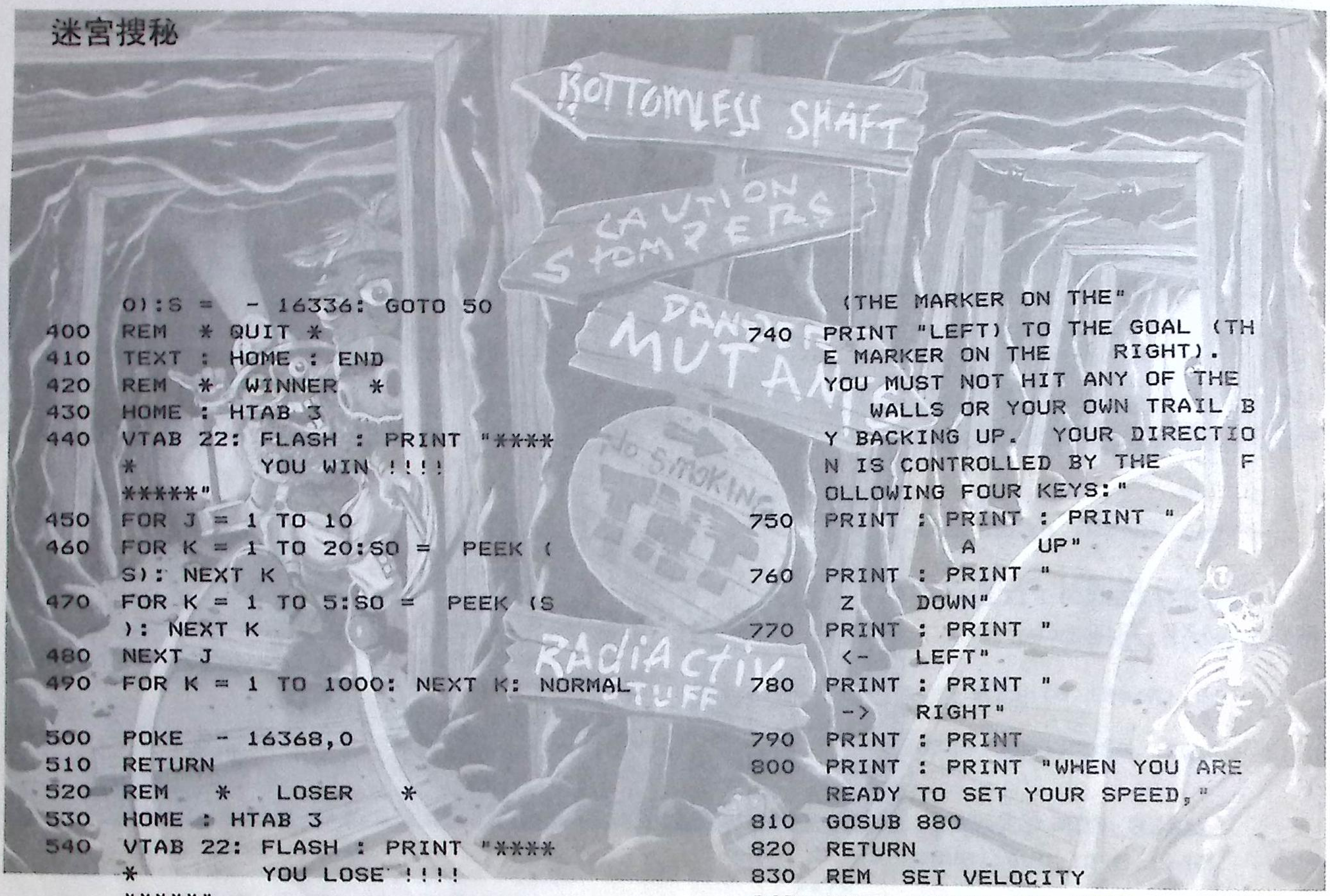
本程式在程式磁碟的 FILENAME 是 MAZE FUN。□



MAZE FUN 程式列表

```
10  REM  MAZE FUN
40  TEXT : HOME : GOSUB 610: REM
     TITLE AND SET UP
50  TEXT : HOME : GOSUB 830: REM
     SET VELOCITY
60  GOSUB 930: REM  DRAW BOARD
70  REM  *   RUN THE MAZE  *
80  COLOR= 13: PLOT X,Y: PLOT XG,
     YG
90  POKE  - 16368,0
100  HOME : VTAB 22: HTAB 15: INVERSE
     : PRINT " SPEED: ";V1;" ": NORMAL
     : VTAB 24: PRINT "PRESS A KE
     Y (A,Z,<-,->) TO START";
110 BE =  PEEK ( - 16384): IF BE <
     127 OR (BE <  > 193 AND BE <
     > 218 AND BE <  > 136 AND B
     E <  > 149) THEN 110
120 BE =  PEEK ( - 16384): FOR J =
     1 TO VE: NEXT J
130  IF BE < 127 OR (BE <  > 193 AND
     BE <  > 218 AND BE <  > 136 AND
     BE <  > 149) THEN BE = LA
140  IF BE = 193 THEN Y = Y - 1: REM
     UP
150  IF BE = 218 THEN Y = Y + 1: REM
     DOWN
160  IF BE = 136 THEN X = X - 1: REM
     LEFT
170  IF BE = 149 THEN X = X + 1: REM
     RIGHT
180 LA = BE
190  IF X = XG AND Y = YG THEN  GOSUB
     430: GOTO 250: REM   WIN
200  IF  SCRN( X,Y) <  > 0 THEN  GOSUB
     530: GOTO 250: REM  LOSE
210  PLOT X,Y
220  FOR J = 1 TO 3:SO =  PEEK (S
     ): NEXT J
230  GOTO 120
240  REM  * CONTINUE OPTIONS *
250  HOME
260  VTAB 21: HTAB 3: PRINT "CHOO
     SE ONE:  <1> SAME MAZE/SAME
     SPEED"
270  VTAB 22: HTAB 16: PRINT "<2>
     SAME MAZE/NEW SPEED"
280  VTAB 23: HTAB 16: PRINT "<3>
     NEW MAZE AND SPEED"
290  VTAB 24: HTAB 16: PRINT "<4>
     QUIT";
300  VTAB 21: HTAB 14: GET K$:K =
     VAL (K$): IF K < 1 OR K > 4
     THEN 300
310  ON K GOTO 330,350,390,410
320  REM  * SAME MAZE & SPEED *
330 X = 4:Y = 36:XG = 36:YG = 4: GOSUB
     1460: GOSUB 1510: GOTO 80
340  REM  * SAME MAZE/NEW SPEED *

350 X = 4:Y = 36:XG = 36:YG = 4
360  TEXT : HOME
370  GOSUB 830: GOSUB 1460: GOSUB
     1510: GOTO 80
380  REM  * NEW MAZE AND SPEED *
390  CLEAR :X = 4:Y = 36:XG = 36:
     YG = 4: DIM M(150): DIM F(15
```



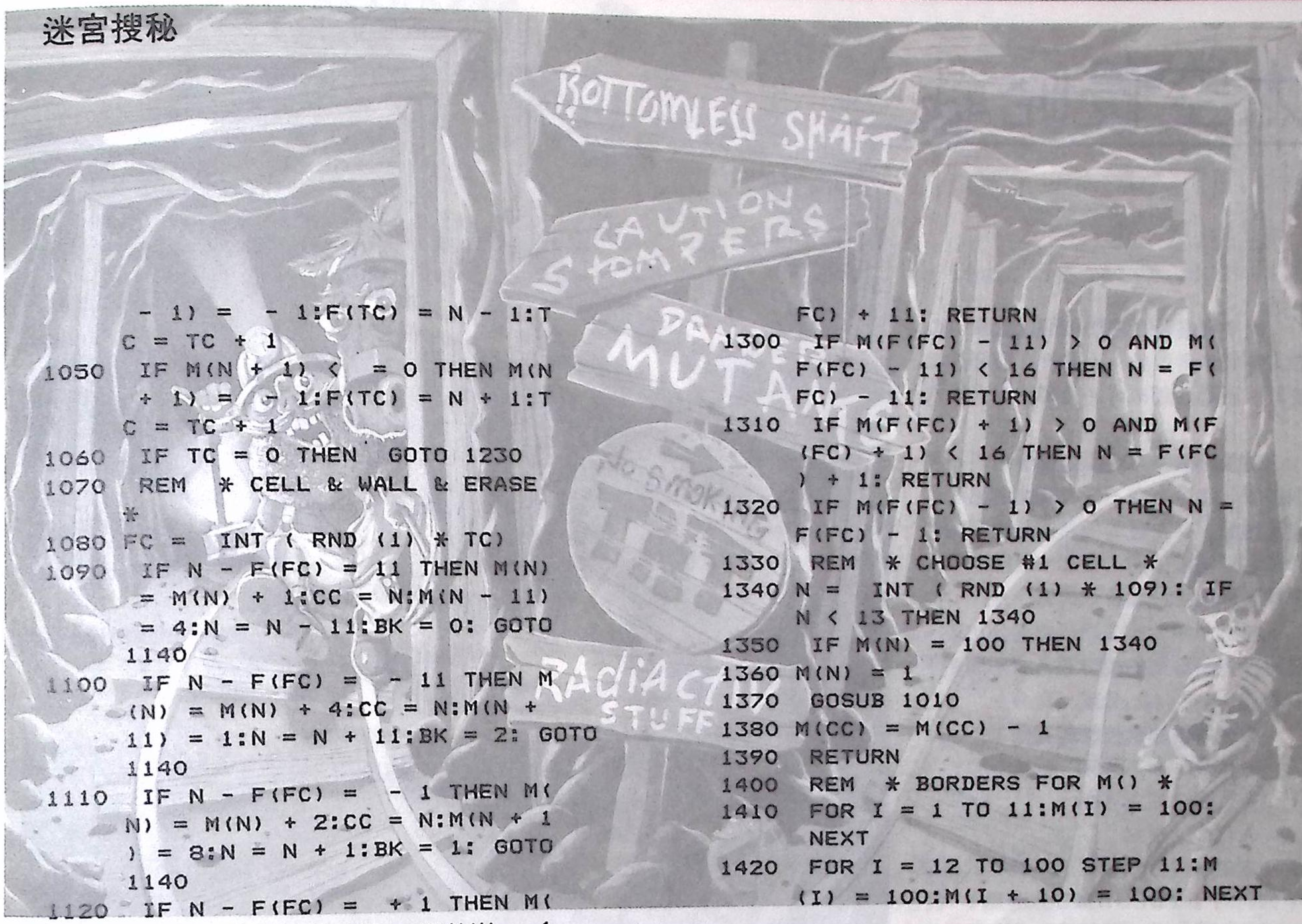

```
    0):S =  - 16336: GOTO 50
400  REM  * QUIT *
410  TEXT : HOME : END
420  REM  *  WINNER  *
430  HOME : HTAB 3
440  VTAB 22: FLASH : PRINT "****
     *        YOU WIN !!!!
     *****"
450  FOR J = 1 TO 10
460  FOR K = 1 TO 20:SO =  PEEK (
     S): NEXT K
470  FOR K = 1 TO 5:SO =  PEEK (S
     ): NEXT K
480  NEXT J
490  FOR K = 1 TO 1000: NEXT K: NORMAL

500  POKE  - 16368,0
510  RETURN
520  REM   *  LOSER   *
530  HOME : HTAB 3
540  VTAB 22: FLASH : PRINT "****
     *        YOU LOSE !!!!
     *****"
550  FOR K = 1 TO 45:SO =  PEEK (
     S): NEXT K
560  FOR K = 1 TO 30: FOR J = 1 TO
     5:SO =  PEEK (S): NEXT J: NEXT
     K
570  FOR J = 1 TO 1000: NEXT J: NORMAL

580  POKE  - 16368,0
590  RETURN
600  REM  * TITLE & SET UP *
610 X = 4:Y = 36:XG = 36:YG = 4: DIM
     M(150): DIM F(150)
620 S =  - 16336
630  HOME : VTAB 5: HTAB 10
640  INVERSE : VTAB 6: HTAB 13: PRINT
     "           "
650  HTAB 13: PRINT "    MAZE   F
     UN    "
660  HTAB 13: PRINT "
         "
670  NORMAL
680  VTAB 20: HTAB 8: PRINT "DO Y
     OU WANT INSTRUCTIONS?";: GET
     K$
690  IF K$ <  > "Y" THEN  RETURN

700  GOTO 720
710  REM  * INSTRUCTIONS *
720  HOME
730  VTAB 1: PRINT "THE OBJECT OF
      SPEED MAZE IS TO THREAD    Y
     OUR WAY THROUGH THE MAZE FRO
     M YOUR      STARTING POSITION
      (THE MARKER ON THE"
740  PRINT "LEFT) TO THE GOAL (TH
     E MARKER ON THE     RIGHT).
     YOU MUST NOT HIT ANY OF THE
         WALLS OR YOUR OWN TRAIL B
     Y BACKING UP.  YOUR DIRECTIO
     N IS CONTROLLED BY THE      F
     OLLOWING FOUR KEYS:"
750  PRINT : PRINT : PRINT "
             A     UP"
760  PRINT : PRINT "
      Z     DOWN"
770  PRINT : PRINT "
      <-    LEFT"
780  PRINT : PRINT "
      ->    RIGHT"
790  PRINT : PRINT
800  PRINT : PRINT "WHEN YOU ARE
     READY TO SET YOUR SPEED,"
810  GOSUB 880
820  RETURN
830  REM  SET VELOCITY
840  HOME : VTAB 12: HTAB 4: INPUT
     "HOW FAST WILL YOU TRAVEL?
                (THE RANGE IS 1
      (SLOW) TO 10 (FAST)      ";V
     $:V1 =  VAL (V$)
850  IF V1 < 1 OR V1 > 10 THEN 84
     0
860 VE = 205 - (V1 * 20)
870  RETURN
880  REM   CONTINUE SEQUENCE
890  VTAB 22: PRINT "        PRESS
     ANY KEY TO CONTINUE         "
900 K =  PEEK ( - 16384): IF K <
     127 THEN 900
910  POKE  - 16368,0: RETURN
920  REM  * DRAW BOARD *
930  GOSUB 1460: REM  DRAW GRID
940  GOSUB 1410: REM  INITIALIZE
     M() ARRAY
950  GOSUB 1340: REM   CHOOSE FIR
     ST CELL
960  GOSUB 1010: REM   ADD ADJ. T
     O FRONTIER
990  GOTO 960
1000  REM  * ADD ADJ. TO FRONT. *

1010 TC = 0
1020  IF M(N - 11) <  = 0 THEN M(
     N - 11) =  - 1:F(TC) = N - 1
     1:TC = TC + 1
1030  IF M(N + 11) <  = 0 THEN M(
     N + 11) =  - 1:F(TC) = N + 1
     1:TC = TC + 1
1040  IF M(N - 1) <  = 0 THEN M(N
```




```
 - 1) =  - 1:F(TC) = N - 1:T
 C = TC + 1
1050  IF M(N + 1) <  = 0 THEN M(N
 + 1) =  - 1:F(TC) = N + 1:T
 C = TC + 1
1060  IF TC = 0 THEN  GOTO 1230
1070  REM  * CELL & WALL & ERASE
 *
1080 FC =  INT ( RND (1) * TC)
1090  IF N - F(FC) = 11 THEN M(N)
 = M(N) + 1:CC = N:M(N - 11)
 = 4:N = N - 11:BK = 0: GOTO
 1140
1100  IF N - F(FC) =  - 11 THEN M
 (N) = M(N) + 4:CC = N:M(N +
 11) = 1:N = N + 11:BK = 2: GOTO
 1140
1110  IF N - F(FC) =  - 1 THEN M(
 N) = M(N) + 2:CC = N:M(N + 1
 ) = 8:N = N + 1:BK = 1: GOTO
 1140
1120  IF N - F(FC) =  + 1 THEN M(
 N) = M(N) + 8:CC = N:M(N - 1
 ) = 2:N = N - 1:BK = 3: GOTO
 1140
1130  REM  * ERASE WALL *
1140 RO =  INT (CC / 11):CO =  INT
 ((CC / 11 -  INT (CC / 11)) *
 11 + .05) - 1
1150  COLOR= 0
1160  IF BK = 0 THEN  HLIN (CO *
 4 - 1),(CO * 4 + 1) AT (RO *
 4 - 2): GOTO 1200
1170  IF BK = 1 THEN  VLIN (RO *
 4 - 1),(RO * 4 + 1) AT (CO *
 4 + 2): GOTO 1200
1180  IF BK = 2 THEN  HLIN (CO *
 4 - 1),(CO * 4 + 1) AT (RO *
 4 + 2): GOTO 1200
1190  IF BK = 3 THEN  VLIN (RO *
 4 - 1),(RO * 4 + 1) AT (CO *
 4 - 2): GOTO 1200
1200  COLOR= HU
1210  RETURN
1220  REM  * CHECK M() FOR FRONT.
 *
1230 TC = 0
1240  FOR I = 13 TO 110
1250  IF M(I) =  - 1 THEN F(TC) =
 I:TC = TC + 1
1260  NEXT I
1270  IF TC = 0 THEN  POP : RETURN

1280 FC =  INT ( RND (1) * TC)
1290  IF M(F(FC) + 11) > 0 AND M(
 F(FC) + 11) < 16 THEN N = F(
 FC) + 11: RETURN
1300  IF M(F(FC) - 11) > 0 AND M(
 F(FC) - 11) < 16 THEN N = F(
 FC) - 11: RETURN
1310  IF M(F(FC) + 1) > 0 AND M(F
 (FC) + 1) < 16 THEN N = F(FC
 ) + 1: RETURN
1320  IF M(F(FC) - 1) > 0 THEN N =
 F(FC) - 1: RETURN
1330  REM  * CHOOSE #1 CELL *
1340 N =  INT ( RND (1) * 109): IF
 N < 13 THEN 1340
1350  IF M(N) = 100 THEN 1340
1360 M(N) = 1
1370  GOSUB 1010
1380 M(CC) = M(CC) - 1
1390  RETURN
1400  REM  * BORDERS FOR M() *
1410  FOR I = 1 TO 11:M(I) = 100:
 NEXT
1420  FOR I = 12 TO 100 STEP 11:M
 (I) = 100:M(I + 10) = 100: NEXT

1430  FOR I = 111 TO 121:M(I) = 1
 00: NEXT
1440  RETURN
1450  REM  * DRAW GRID *
1460  HOME : GR : READ HU: IF HU =
 100 THEN  RESTORE : GOTO 146
 0
1470  COLOR= HU
1480  DATA  6,9,7,14,11,12,3,100

1490  FOR I = 2 TO 38 STEP 4: HLIN
 2,38 AT I: VLIN 2,38 AT I: NEXT
 : RETURN
1500  REM  * REDRAW SAME MAZE *
1510  COLOR= 0: FOR I = 13 TO 109
 : IF M(I) = 100 THEN  NEXT
1520 ER = M(I):RO =  INT (I / 11)
 :CO =  INT ((I / 11 -  INT (
 I / 11)) * 11 + .05) - 1
1530  IF ER > 7 THEN ER = ER - 8:
 VLIN (RO * 4 - 1),(RO * 4 +
 1) AT (CO * 4 - 2)
1540  IF ER > 3 THEN ER = ER - 4:
 HLIN (CO * 4 - 1),(CO * 4 +
 1) AT (RO * 4 + 2)
1550  IF ER > 1 THEN ER = ER - 2:
 VLIN (RO * 4 - 1),(RO * 4 +
 1) AT (CO * 4 + 2)
1560  IF ER = 1 THEN  HLIN (CO *
 4 - 1),(CO * 4 + 1) AT (RO *
 4 - 2)
1570  NEXT I
1580  COLOR= HU: RETURN
```
■

太空侵略戰



太空侵略戰是模仿市面遊戲機中心的太空侵略者遊戲，不過沒有那麼複雜設計。不過玩起來時，也相當令人投入。

遊戲開始時，螢幕底部會出現一隻保衛者，這是上空不斷有侵略者出現欲降落地面，保衛者的任務就是要在他們降落之前消滅他們，否則一旦降落在水平線，保衛者反而會被消滅。

遊戲中總共有3類侵略者會出現，殲滅他們所得到的獎分是各有不同的，有些 30 分、40 分或 50 分。

保衛者的移動控制相當簡單：

→代表　右方移動

←代表　左方移動

A代表　發炮

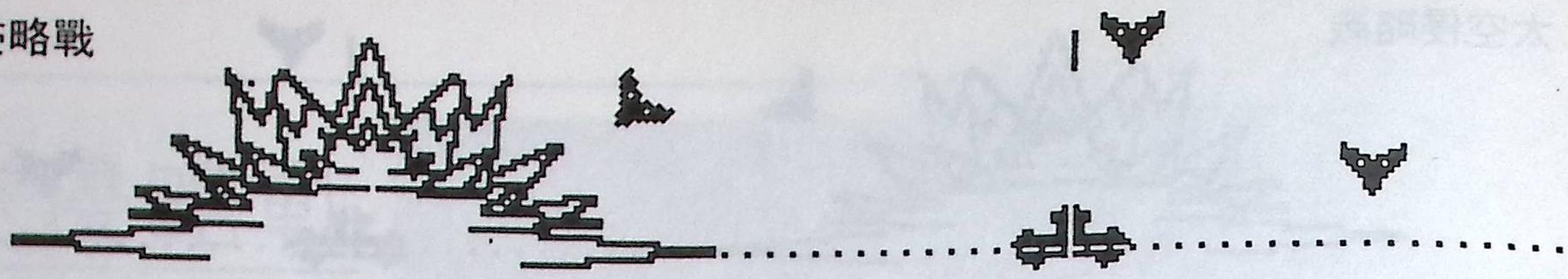
本遊戲同分兩個程式，在程式磁碟中他們的 FILENAME 分別爲：（1）SPACE ATTACK；（2）SPACE SHIP。

在 RUN 過程式後，全自動產生一個記分 FILE（在程式磁碟中爲 SPACE SCORE）。□

```
SPACE SHIP
1DD8- 11 00 2F 00 65 00 79 00
1DE0- 98 00 B4 00 CD 00 EB 00
1DE8- F9 00 01 01 0B 01 17 01
1DF0- 21 01 2D 01 38 01 41 01
1DF8- 4C 01 6F 01 00 00 00 00
1E00- 00 00 00 00 00 00 00 36
1E08- 36 36 3F 3F 3F 3F 27 29
1E10- 2C 2E 2D 2D 04 24 3C 2E
1E18- 09 21 34 36 36 2E 2D 2D
1E20- 2D 2D 1C 27 37 3F 3F 07
1E28- 24 24 35 07 00 FF 00 00
1E30- 00 00 00 00 00 00 00 00
1E38- 00 00 00 00 00 2D 2D 2D
1E40- 2D 37 3F 2F 3E 1B 1B 18
1E48- 18 3F 3F 3F 3F 35 2D 3D
1E50- 2E 00 12 02 36 36 03 24
1E58- 24 3C 27 3F 34 2E 2E 2E
1E60- 2E 2D 36 24 24 2C 25 2D
1E68- 34 3E 3E 3E 3E 37 36 04
1E70- 00 3F 3E 3E 3E 36 25 2D
1E78- 24 25 2D 3E 3E 36 2F 25
1E80- 2D 24 2E 36 37 37 37 27
1E88- 37 3C 3C 04 00 2F 3D 36
1E90- 3F 3C 27 2E 2E 2E 2D 2C
1E98- 2C 25 3E 3E 3E 37 36 36
1EA0- 37 2F 2D 3D 34 00 3F 3A
1EA8- 3A 32 0D 0D 16 16 3F 38
1EB0- 38 20 09 09 09 0D 36 33
1EB8- 33 3B 04 20 20 20 28 15
1EC0- 15 15 36 00 0C 25 1C 3F
1EC8- 17 36 2E 1E 0E 2D 05 24
1ED0- 24 00 24 BC 96 12 2D 1C
1ED8- 24 00 65 E4 3F 17 96 F1
1EE0- 2E 2D 25 00 0C 0C 3C 3F
1EE8- B7 92 15 2D 0C E4 07 00
1EF0- 3A 27 0C 0C 0C 36 AE 37
1EF8- 3E 00 38 27 2C 2D F5 AA
1F00- 36 17 3F 1C 04 00 75 F6
1F08- 3F 1C 24 E5 0C 0C 2D 06
1F10- 00 0C 0C 3C 3F B7 52 1E
1F18- 2E 00 E7 64 2D 15 F6 0E
1F20- F6 3F 1C 24 00 E7 64 2D
1F28- 15 36 77 1E 1E 3F 04 00
1F30- E7 64 2D 15 F6 0E F6 3F
1F38- 1C 24 00 E7 64 2D 15 36
1F40- 77 1E 1E 3F 04 00 24 34
1F48- 36 36 36 00 FF FF 00 00
1F50- 24 34 36 36 36 00 00 00
1F58- 00 00 00 00 00 00 00 00
1F60- 00 00 00 00 00 00 00 00
1F68- 20 B9 1F A9 00 8D C2 20
1F70- A9 90 85 05 4C BD 1E A9
1F78- 70 20 68 3D 20 50 20 20
1F80- 8E FD 4C 6E 3A A9 71 20
1F88- 68 3D 20 8E FD 4C 6E 3A
1F90- 48 20 05 38 68 AA A0 02
1F98- B1 0D 29 7F 9D BF 20 60
1FA0- 20 05 38 A0 00 B1 0D 8D
1FA8- 8A 39 C8 B1 0D 8D 8B 39
1FB0- A9 01 8D 92 39 20 18 20
1FB8- 60 A9 00 8D BD 20 8D B8
1FC0- 20 A9 01 8D 92 39 AD BE
1FC8- 20
```



```
20  REM  ************************
30  REM  *                      *
40  REM    *   SPACE  ATTACK
      *
50  REM  *                      *
60  REM  ************************
90  LOMEM: 24576: GOSUB 1440: GOTO
     1130
100  HOME : VTAB 12: PRINT "**
     SPACE ATTACK**       ": GOSUB
     1300
110 D$ =  CHR$ (4)
120  PRINT D$"BLOAD SPACE SHIP": POKE
     216,0
130  DIM FX(25): DIM FY(25): DIM
     FT(25)
140  DATA  255,90,190,90,151,20,1
     90,20,255,90,151,90,190,90,1
     51,20,190,20,255,90,151,90,1
     90,90,151,20,190,20,255,90,2
     55,90,190,200
150  GOSUB 1200
160 NF = 5:MF = 5
170  RESTORE
180 GM =  FRE (0)
190  HGR2 : HCOLOR= 3: ROT= 0: SCALE=
     1
200  POKE 232,216: POKE 233,29
210  GOSUB 1280
220  GOSUB 240
230  GOTO 450
240  HPLOT 0,1 TO 0,3: HPLOT 1,0 TO
     1,4 TO 2,4 TO 2,3 TO 3,3 TO
     3,7: HPLOT 4,4 TO 4,6: HPLOT
     3,7 TO 0,7 TO 0,6 TO 1,6
250  HPLOT 139,0 TO 139,7: HPLOT
     140,0 TO 140,7: HPLOT 142,0 TO
     142,7: HPLOT 143,0 TO 143,7:
      HPLOT 141,3: HPLOT 147,0 TO
     147,7: HPLOT 146,0 TO 146,7
260  HPLOT 150,1 TO 150,3: HPLOT
     151,0 TO 151,4 TO 152,4 TO 1
     52,3 TO 153,3 TO 153,7: HPLOT
     154,4 TO 154,6: HPLOT 153,7 TO
     150,7 TO 150,6 TO 151,6
270  HPLOT 2,0 TO 4,0 TO 1,4 TO 1
     ,3
280  HPLOT 152,0 TO 154,0 TO 151,
     4 TO 151,3
290  HPLOT 7,1 TO 7,6: HPLOT 8,0 TO
     8,7 TO 11,7 TO 11,6 TO 10,6
300  HPLOT 157,1 TO 157,6: HPLOT
     158,0 TO 158,7 TO 161,7 TO 1
     61,6 TO 160,6
310  HPLOT 8,0 TO 11,0 TO 11,1 TO
     10,1
320  HPLOT 158,0 TO 161,0 TO 161,
     1 TO 160,1
330  HPLOT 14,1 TO 14,6: HPLOT 15
     ,0 TO 15,7: HPLOT 16,0: HPLOT
     16,7
340  HPLOT 164,1 TO 164,6: HPLOT
     165,0 TO 165,7: HPLOT 166,0:
      HPLOT 166,7
350  HPLOT 18,1 TO 18,6: HPLOT 17
     ,0 TO 17,7
360  HPLOT 168,1 TO 168,6: HPLOT
     167,0 TO 167,7
370  HPLOT 21,0 TO 21,7 TO 22,7 TO
     22,0 TO 24,0 TO 24,7 TO 25,7
      TO 25,4 TO 23,4 TO 23,3 TO
     24,3 TO 25,2 TO 25,1: HCOLOR=
     0: HPLOT 25,4: HCOLOR= 3
380  HPLOT 171,0 TO 171,7 TO 172,
     7 TO 172,0 TO 174,0 TO 174,7
      TO 175,7 TO 175,4 TO 173,4 TO
     173,3 TO 174,3 TO 175,2 TO 1
     75,1: HCOLOR= 0: HPLOT 175,4
     : HCOLOR= 3
390  HPLOT 28,1 TO 28,6: HPLOT 29
     ,0 TO 29,7 TO 32,7 TO 32,6 TO
     31,6
400  HPLOT 178,1 TO 178,6: HPLOT
     179,0 TO 179,7 TO 182,7 TO 1
     82,6 TO 181,6
410  HPLOT 29,0 TO 32,0 TO 32,1 TO
     31,1: HPLOT 30,3 TO 31,3
420  HPLOT 179,0 TO 182,0 TO 182,
     1 TO 181,1: HPLOT 180,3 TO 1
     81,3
430  HPLOT 35,2 TO 36,2 TO 36,3 TO
     35,2: HPLOT 35,5 TO 36,5 TO
     36,6 TO 35,5
440  HPLOT 185,2 TO 186,2 TO 186,
     3 TO 185,2: HPLOT 185,5 TO 1
     86,5 TO 186,6 TO 185,5: RETURN
450  FOR I = 0 TO 272 STEP 4: HPLOT
     I,160: NEXT
460  FOR MK = 1 TO 17: IF  PEEK (
     - 16384) = 77 + 128 THEN MK
     = 17
470  READ NM: READ MN: POKE 6,NM:
     POKE 7,MN: CALL 768: NEXT
480 SC = 0
490 PF = 15:FF = 0
500 X = 140:F = 0:NW = 0:DC = 1
```



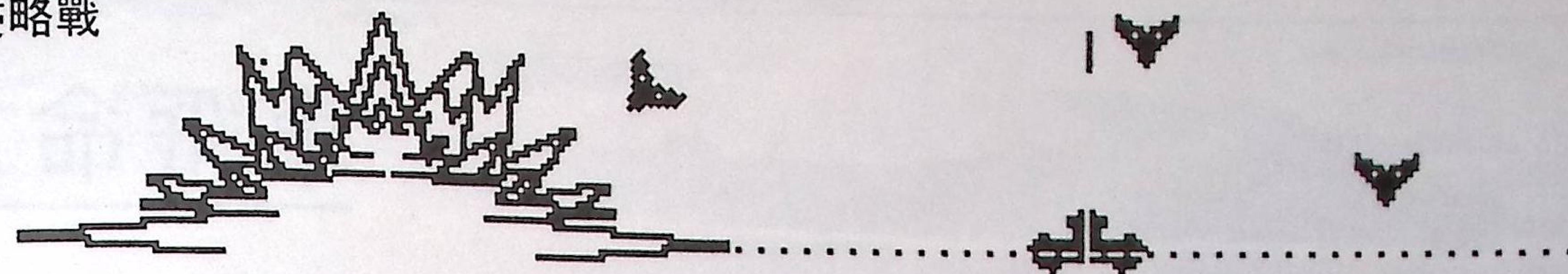
```
510 X = 140
520 NW = NW + 1:FF = FF + 1: IF F
     F = 4 THEN FF = 1
530 MF = MF + 1:NF = MF
540 NL = 1:
550  FOR I = 1 TO NF:FX(I) =  INT
     ((25 *  RND (1))):FX(I) = FX
     (I) * 10 + 20:FY(I) = 15: NEXT

560  FOR I = 1 TO NF:FT(I) = 20 -
     I - NW: IF FT(I) < 3 THEN FT
     (I) = 3
570  NEXT
580 FT(NF) = 50
590  FOR.XX = 1 TO NF
600  FOR I = 1 TO FT(NL)
610  IF I = 35 AND FT(NL) = 50 THEN
      HCOLOR= 3:MSI = MSI: DRAW F
     F + 2 AT PF,180:PF = PF + 15
     : IF PF = 270 THEN  HCOLOR=
     0: FOR Y = 179 TO 191: HPLOT
     0,Y TO 279,Y: NEXT :PF = 15
620  FOR O = 1 TO NL - 1
630  IF FY(O) = 200 THEN 700
640  HCOLOR= 0: DRAW FF + 2 AT FX
     (O),FY(O)
650 FY(O) = FY(O) + 3
660  HCOLOR= 3
670  DRAW FF + 2 AT FX(O),FY(O)
680  IF F = 1 THEN  IF FX(O) = BX
      AND BY < FY(O) + 10 AND BY >
     FY(O) - 13 THEN  HCOLOR= 0: DRAW
     FF + 2 AT FX(O),FY(O): DRAW
     17 AT BX,BY: HCOLOR= 3: DRAW
     6 AT FX(O),FY(O): HCOLOR= 0:
      DRAW 6 AT FX(O),FY(O):F = 0
     :FY(O) = 200: GOSUB 1240: GOSUB
     1160: GOTO 700
690  IF FY(O) =  > 153 THEN  GOSUB
     880
700  NEXT O
710  REM  *********************
720  HCOLOR= 0: DRAW 1 AT X,153: DRAW
     2 AT X + 2,161
730  IF  PEEK (9000) = 50 THEN 78
     0
740  IF  PEEK ( - 16384) = 149 AND
     X < 261 THEN X = X + 10
750  IF  PEEK ( - 16384) = 136 AND
     X > 21 THEN X = X - 10
760  IF F = 0 THEN  IF  PEEK ( -
     16384) = 193 THEN F = 1:BX =
     X:BY = 162: POKE  - 16368,0
770  GOTO 810
780  IF  PDL (0) > 254 AND X < 26
     1 THEN X = X + 10
790  IF  PDL (0) < 6 AND X > 21 THEN
     X = X - 10
800  IF F = 0 THEN  IF  PEEK ( -
     16287) > 127 THEN F = 1:BX =
     X:BY = 162
810  HCOLOR= 3: DRAW 1 AT X,153: DRAW
     2 AT X + 2,161
820  IF F = 1 THEN  HCOLOR= 0: DRAW
     17 AT BX,BY:BY = BY - 21: IF
     BY < 22 THEN F = 0
830  IF F = 1 THEN  HCOLOR= 3: DRAW
     17 AT BX,BY
840  NEXT I:NL = NL + 1
850  NEXT XX
860  GOTO 520
870  REM
880  HCOLOR= 0: DRAW FF + 2 AT FX
     (O),FY(O): GOSUB 1230
890 M = FX(O):P = FX(O): FOR PP =
     0 TO 272:M = M - 1:P = P + 1

900  IF M =  > 0 THEN  HPLOT M,16
     0
910  IF P =  < 272 THEN  HPLOT P,
     160
920  IF M < X AND P > X THEN PP =
     272
930  IF M < 0 AND P > 272 THEN PP
      = 272
940  NEXT
950  DRAW 1 AT X,153: DRAW 2 AT X
      + 2,161:XX = 20
960  FOR Z = 153 TO 170: HCOLOR=
     0: DRAW 1 AT X,Z - 1: DRAW 2
      AT X + 2,Y + 7
970  HCOLOR= 3: DRAW 1 AT X,Z: DRAW
     2 AT X + 2,Z + 8: FOR I = 1 TO
     XX / 2: NEXT I:XX = XX - 1: NEXT

980  HCOLOR= 0: DRAW 1 AT X,Z: DRAW
     2 AT X + 2,Z + 8: HCOLOR= 3
990  FOR O = 1 TO 3
1000  FOR LL = 1 TO 10: POKE 6,0:
     POKE 7,1: CALL 768: NEXT
1010 Y = 173
1020  IF X > 59 THEN  HPLOT X - 9
      * O,Y + 6 * O TO X - 20 * O
     ,Y + 5 * O TO X - 10 * O,Y +
     4 * O TO X - 13 * O,Y + 2 *
     O TO X - 8 * O,Y + 3 * O
1030  FOR LL = 1 TO 10: POKE 6,0:
     POKE 7,1: CALL 768: NEXT
1040  IF X > 59 THEN  HPLOT  TO X
     - 11 * O,Y - 1 * O TO X - 7
      * O,Y + 1 * O TO X - 8 * O,
     Y - 5 * O TO X - 6 * O,Y - 2
      * O TO X - 5 * O,Y - 5 * O TO
     X - 2 * O,Y - 2 * O TO X,Y -
     7 * O
1050  FOR LL = 1 TO 10: POKE 6,0:
     POKE 7,1: CALL 768: NEXT
1060  IF X < 219 THEN  HPLOT X +
     9 * O,Y + 6 * O TO X + 20 *
     O,Y + 5 * O TO X + 10 * O,Y +
```

太空侵略戰



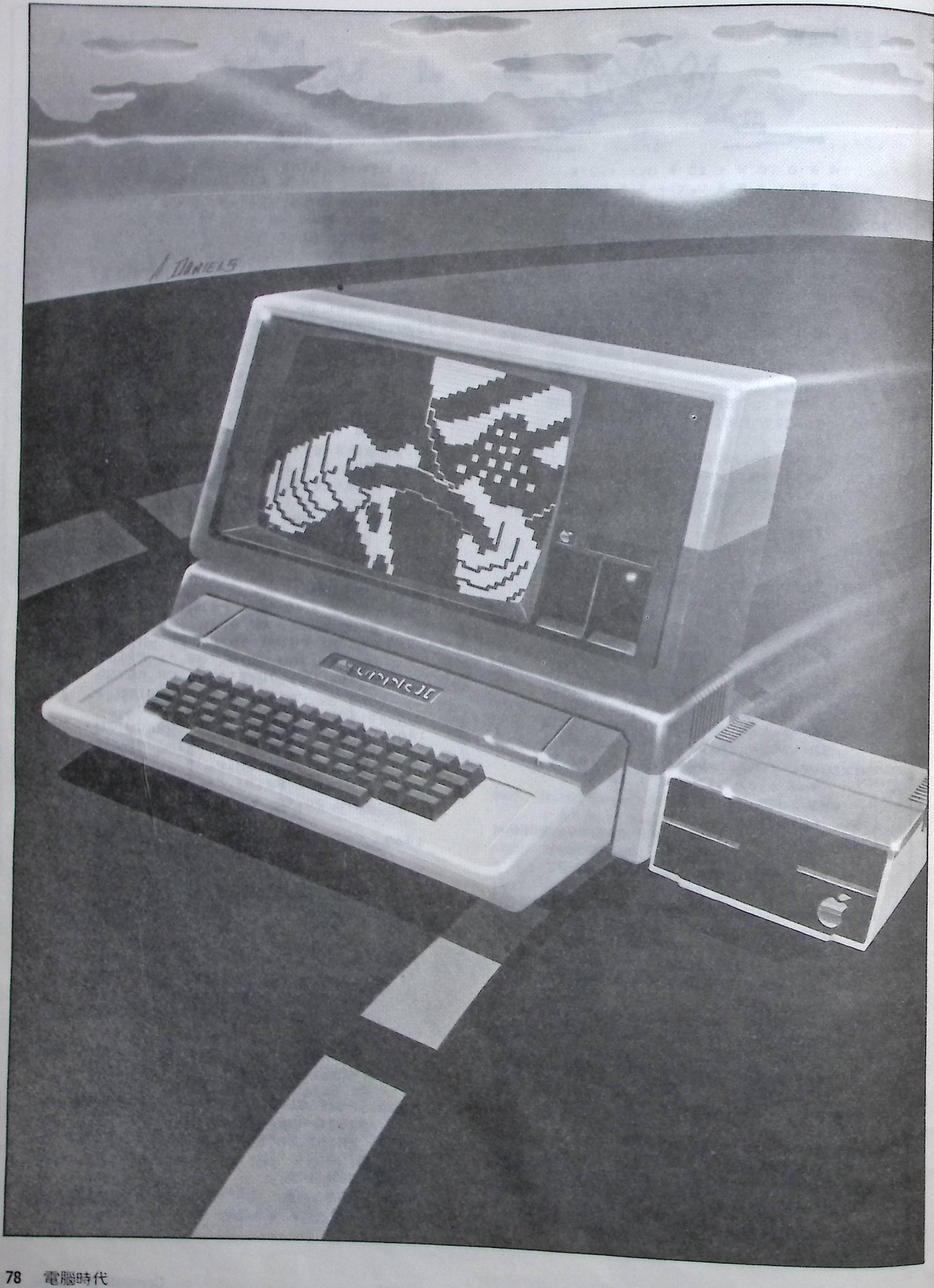
```
     4 * O TO X + 13 * O,Y + 2 *
     O TO X + 8 * O,Y + 3 * O TO
     X + 11 * O,Y - 1 * O
1070  FOR LL = 1 TO 10: POKE 6,0:
     POKE 7,1: CALL 768: NEXT
1080  IF X < 219 THEN  HPLOT  TO
     X + 7 * O,Y + 1 * O TO X + 8
      * O,Y - 5 * O TO X + 6 * O,
     Y - 2 * O TO X + 5 * O,Y - 5
      * O TO X + 2 * O,Y - 2 * O TO
     X,Y - 7 * O
1090  NEXT O
1100  GOSUB 1360
1110 : HOME : TEXT : VTAB 8: PRINT
     "SCORE:";SC: VTAB 10: GOSUB
     1460
1120  POKE  - 16368,0
1130  GET A$: IF A$ = "P" OR A$ =
     "K" THEN  POKE 9000,(50 * (A
     $ = "P") + 10 * (A$ = "K")):
      RUN 100
1140  IF A$ = "Q" THEN  HOME : END

1150  GOTO 1130
1160  HCOLOR= 0:SC$ =  STR$ (SC):
     SR =  LEN (SC$): FOR W = 1 TO
     SR:WW$ =  MID$ (SC$,W,1):WW =
      VAL (WW$): DRAW WW + 7 AT W
      * 7 + 37,5: NEXT
1170 SC = SC + 10 * (FF + 2)
1180  HCOLOR= 3:SC$ =  STR$ (SC):
     SR =  LEN (SC$): FOR W = 1 TO
     SR:WW$ =  MID$ (SC$,W,1):WW =
      VAL (WW$): DRAW WW + 7 AT W
      * 7 + 37,5: NEXT : IF SC >
     HS THEN HS = SC
1190  RETURN
1200  POKE 768,173: POKE 769,48: POKE
     770,192: POKE 771,136: POKE
     772,208: POKE 773,4: POKE 77
     4,198: POKE 775,7
1210  POKE 776,240: POKE 777,8: POKE
     778,202: POKE 779,208: POKE
     780,246: POKE 781,166: POKE
     782,6: POKE 783,76: POKE 784
     ,0: POKE 785,03: POKE 786,96

1220  RETURN
1230  FOR FD = 1 TO 3: POKE 6,200
     : POKE 7,30: CALL 768: POKE
     6,80: POKE 7,30: CALL 768: NEXT
     : RETURN
1240 SN =  INT (50 *  RND (1)) +
     1
1250 ND =  INT (50 *  RND (1)) +
     1
1260  FOR O = 1 TO 7: POKE 6,SN: POKE
     7,1: CALL 768: POKE 6,ND: POKE
     7,1: CALL 768: NEXT
1270  RETURN
1280 HS$ =  STR$ (HS):SR =  LEN (
     HS$): FOR W = 1 TO SR:WW$ =
      MID$ (HS$,W,1):WW =  VAL (W
     W$): DRAW WW + 7 AT W * 7 +
     190,5: NEXT
1290  RETURN
1300 D$ =  CHR$ (4): ONERR  GOTO
     1490
1310  PRINT D$
1320  PRINT D$"BLOAD HISCORE": POKE
     216,0
1330  FOR I = 1 TO  PEEK (8195)
1340 NC =  PEEK (8195 + I):HS$ =
     HS$ +  CHR$ (NC): NEXT
1350 HS =  VAL (HS$): RETURN
1360 HS$ =  STR$ (HS)
1370  POKE 8195, LEN (HS$)
1380  FOR O = 1 TO  LEN (HS$)
1390  POKE 8195 + O, ASC ( MID$ (
     HS$,O,1)): NEXT
1400 D$ =  CHR$ (4)
1410  PRINT D$
1420  PRINT D$"BSAVE HISCORE,A819
     5,L8"
1430  RETURN
1440  TEXT : HOME : PRINT "SUCCES
     SIVE WAVES OF EVIL SHAPE TAB
     LES ": PRINT "WILL DESCEND D
     OWN ON YOU WITH THE ": PRINT
     "PADDLES, JOYSTICK, OR KEYBO
     ARD ": PRINT "YOUR MISSION I
     S TO DESTROY THEM "
1445  PRINT "BEFORE THEY REACH TH
     E ENERGY LINE ": PRINT "SUSP
     ENDING YOUR SHIP. "
1450  PRINT : PRINT "ALIENS ARE W
     ORTH 30,40 OR 50 POINTS": PRINT

1460  PRINT "PRESS:": PRINT : PRINT
     "   P-FOR PADDLE/JOYSTICK CO
     NTROL": PRINT "   K-FOR KEYB
     OARD CONTROL": PRINT "   (AR
     ROWS MOVE LEFT/RIGHT)": PRINT
     "   ('A' KEY FIRES AND STOPS
     )": PRINT "   Q-TO QUIT"
1470  PRINT : PRINT : PRINT : PRINT
     "(PRESS 'M' DURING MUSIC FOR
      SILENCE)": PRINT "      **S
     PACE ATTACK**": VTAB 16: HTAB
     14
1480  RETURN
1490  CALL  - 10621: IF  PEEK (22
     2) <  > 6 THEN 1520
1500  POKE 8195,1: FOR R = 8196 TO
     8202: POKE R,0: NEXT
1510  PRINT D$"BSAVE HISCORE,A819
     5,L8"
1520  RUN 100
4040  REM  *    SPACE ATTACK
      *
```

■



亡命飛車

本程式是一項賽車的遊戲。車子在一條曲折的道路上飛馳，但車子會撞着道路上的障礙物，而失去一次機會。當車子繼續向前奔馳時，道路開始變得窄而且彎曲，但假如車子能夠停留在道路上，就可以得到高分子。

這個遊戲共有4級難度，由一至四；第一級是初學者的，第二級是速度快。而道路彎曲，第三級是有障礙物的，而第四級就是有障礙物而速度快。

當車子撞着路邊或者障礙物時，你都可以聽到三次爆炸聲。而道路的濶度，和你所得的分數都會在畫面上出現。車子必須用 JOYSTICK 來控制的。

本程式在磁碟中的 FILENAME 是 ROADER。□

ROADER 程式

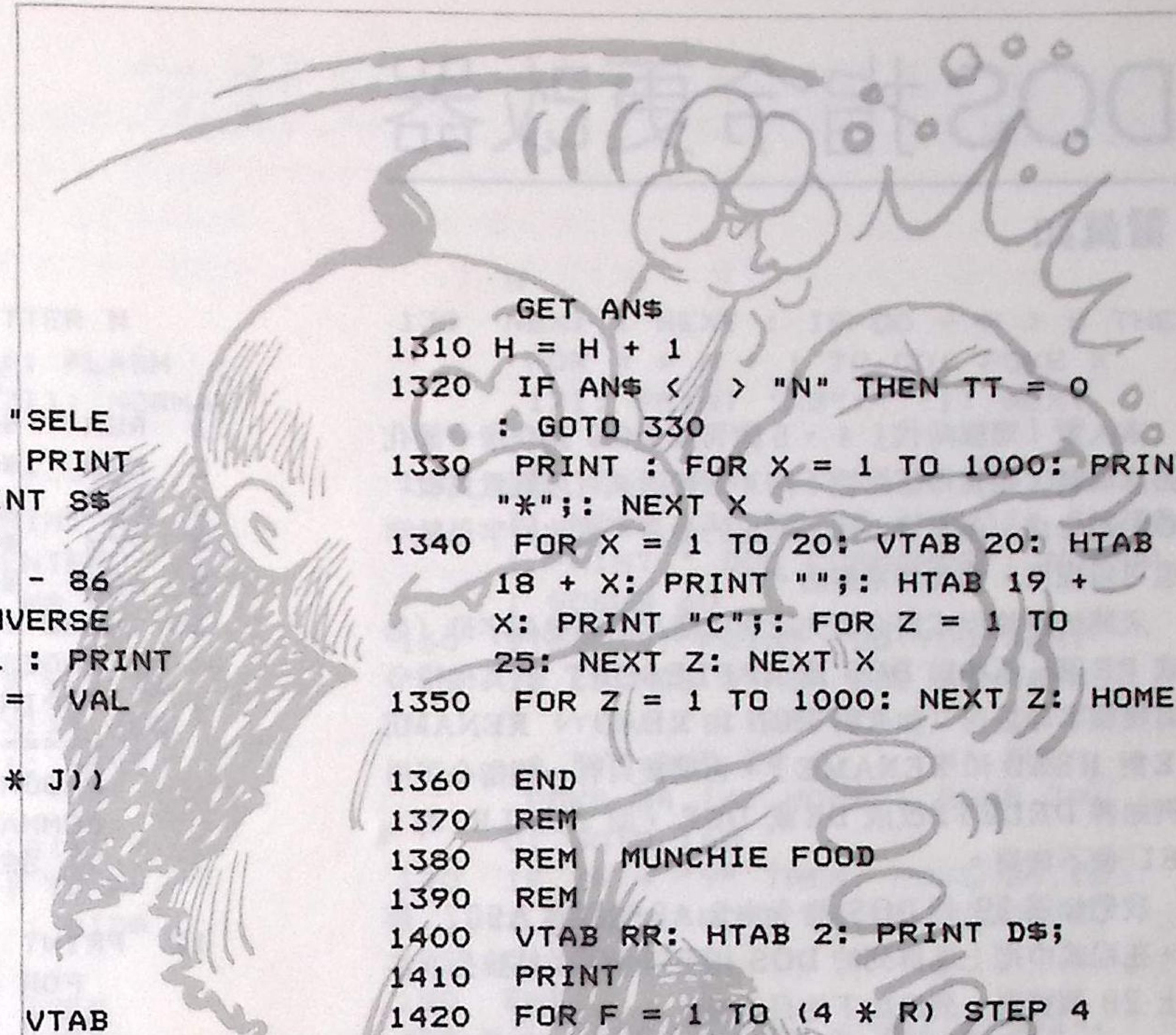
```
100 N$ = " REDAOR":D = 0:A = 0:B =
     0
110  HOME
120  FOR I = 1 TO 7:N$(I) =  MID$
     (N$,I,1): NEXT I
130  FOR I = 1 TO 7:A = A + .4:N =
      INT ( COS (A) * 8)
140  VTAB 24 - D - I: HTAB 20 + N
     : PRINT N$(I)
150  NEXT I:B = B + .4:A = B: IF
     D = 16 THEN 170
160 D = D + 1: GOTO 130
170  VTAB 12: PRINT " WHAT SKILL
     LEVEL DO YOU WISH TO PLAY? "
180  PRINT : PRINT "1) EASY ";: HTAB
     26: PRINT "2) INTERMEDIATE "
190  PRINT "3) DIFFICULT ";: HTAB
     26: PRINT "4) EXPERT "
200  PRINT : PRINT " USE PADDLE 0
      TO CONTROL YOUR CAR. ": PRINT
210  GET LV$:LV =  VAL (LV$)
220  IF LV < 1 OR LV > 4 THEN 210
230 C = 0: IF LV = 2 OR LV = 4 THEN
     C = .05
240 C = C + .05:C1 = 14:C2 = 25:A
     = 0:SC = 8
250 N$ = "
                                 "
260  HOME
270 A = A + C + LV / 16:Y =  INT
     ( COS (A) * 10)
280  POKE YLOC,160: INVERSE
290  PRINT  LEFT$ (N$,C1 + Y);: PRINT
     "^";: HTAB C2 + Y: PRINT "^"
     ;: PRINT  LEFT$ (N$,C1 - Y +
     1);
300 N =  INT ( PDL (0) / 7):XLOC =
     N + 1360
310  IF  PEEK (XLOC) = 30 OR  PEEK
     (XLOC) = 42 OR  PEEK (XLOC) =
     32 THEN 380
320  NORMAL : POKE XLOC,200:YLOC =
     XLOC
330  IF LV = 1 OR LV = 2 THEN 350
340 G =  INT ( RND (1) * 120): IF
     G = 1 THEN G =  INT ( RND (1
     ) * 39): POKE 1872 + G,42
350 D = D + 1: IF D / 120 =  INT
     (D / 120) AND D < 480 THEN C
     1 = C1 + 1:SC = SC - 1
360  IF C < .25 THEN C = C + .001
370  GOTO 270
380  FOR I = 1 TO 20
390  FOR C = 1 TO 15:W =  PEEK ( -
     16336): NEXT C
400  POKE XLOC - 128,220: POKE XL
     OC - 128,225: POKE XLOC - 12
     8,239: POKE XLOC - 128,223
410  NEXT I: NORMAL
420  HOME : VTAB 5: HTAB 10: PRINT
     "YOUR SCORE  IS ";: INVERSE
     : PRINT  INT (10000 / SC): NORMAL
430  VTAB 10: PRINT "HIT THE PADD
     LE BUTTON TO PLAY AGAIN AT":
      HTAB 10: PRINT "THE SAME LE
     VEL,OR:": PRINT
440  PRINT "TYPE(S) TO START OVER
     ,(E) TO END."
450  IF  PEEK ( - 16384) = 197 THEN
      POKE  - 16287,0: END
460  IF  PEEK ( - 16384) = 211 THEN
      POKE  - 16287,0: GOTO 100
470  IF  PEEK ( - 16287) > 127 THEN
     230
480  GOTO 450
```
■

累加遊戲

這是一個簡單而有趣的遊戲。當遊戲開始時，螢幕會出示八列星星及在右邊有相當於星星數目的數字，這些數字是會改變的。你只需選擇其中一行（按照左邊的數字）每選一行，數目便會累加起來，如發現加數等於八列中其中任何未選定的一列，可按 9 選擇便結束，遊戲此時會詢問你要的總數。

每一行的數目不會大於 8，所以當所加的數目大過 8，你便算輸了。當你選擇某行後，它便會變成 0，而你便不可以再選擇這行，否則也當輸。□

```
11  REM    *     MUNCHIES     *
12  REM
13  REM
100  TEXT : HOME : CLEAR : REM  3
     K
220  REM
230  REM  INITIALIZE
240  REM
250 M$ = "MUNCHIE"
260 Q$ =  CHR$ (7)
270 S$ =  CHR$ (32)
280 B$ =  CHR$ (8) +  CHR$ (8)
290  PRINT "NEED INSTRUCTIONS (Y/
     N)";B$;: GET AN$
300  IF AN$ = "Y" THEN  GOSUB 145
     0
310  FOR X = 1 TO 38:D$ = D$ +  CHR$
     (45): NEXT X
320 H = 1
330  HOME
340  REM
350  REM  SET UP TRACK
360  REM
370  VTAB 12: HTAB 16: INVERSE : PRINT
     "MUNCHIES ";H: NORMAL
380  FOR Y = 1 TO 8
390  VTAB Y: HTAB 1: PRINT D$
400 R =  INT (8 *  RND (1)) + 1
410  VTAB Y: HTAB 1: PRINT Y;
420  VTAB Y: HTAB 39: PRINT R
430  NEXT Y
440  VTAB 10: HTAB 5: PRINT "==PR
     ESS(9) FOR TARGET SCORE=="
450  REM
460  REM  RANDOMIZE POINTS
470  REM
480  FOR B = 1 TO 8
490  FOR RR = 1 TO 8
500 R =  INT (8 *  RND (1)) + 1
510  REM
520  REM  ZERO CATCHER
530  REM
540  IF  PEEK (934 + (RR * 128)) =
     176 THEN 570
550  GOSUB 1370
560  VTAB RR: HTAB 39: PRINT R
570  NEXT RR
580  REM
590  REM  SELECTOR
600  REM
610  FOR X = 10 TO 22: VTAB X - 1
     : HTAB 39: PRINT S$;: VTAB X
     : CALL  - 868: HTAB 39: PRINT
     "C";: FOR Z = 1 TO 20: NEXT
     Z: NEXT X
620  IF K <  > 0 THEN M$ = "C"
630  SPEED= 176: FOR I = 38 TO 1 STEP
      - 1: VTAB 22: HTAB 1: PRINT
     M$: NEXT I:K = 1
640  VTAB 22: HTAB 1: PRINT "WHIC
     H LINE?";: GET A$:A =  VAL (
     A$): SPEED= 255
650  IF A = 0 THEN 940
660  IF A = 9 THEN 990
670  INVERSE : VTAB A: HTAB 1: PRINT
     A;: NORMAL
680  FOR X = 2 TO 36
690  VTAB A: HTAB X: PRINT S$;L$;
     : FOR Z = 1 TO 50: NEXT Z
700  REM
710  REM  MUNCHIE MAKER
720  REM
730  IF  INT (X / 2) = X / 2 THEN
     L$ = "=": GOTO 750
740 L$ = "C"
750  GOSUB 1130
760  NEXT X
770  POKE  - 16368,0
780 P =  PEEK (934 + (A * 128)):P
     P = P - 176
790  IF PP = 0 THEN 1180
800 TT = TT + PP
810  IF TT > 8 THEN 1220
820  POKE (934 + (128 * A)),176
830  POKE (933 + (128 * A)),190: PRINT
     Q$;
840  REM
850  REM  SCOREBOARD
860  REM
870  PRINT
880  VTAB 19: PRINT  TAB( 1)"LAST
     RUN ";PP; TAB( 15)"TOTAL ";
     : INVERSE : PRINT TT;: NORMAL
     : PRINT  TAB( 25)"TEXT TURN
     NO.";B + 1
890 A = 0
900  NEXT B
910  REM
920  REM  SUDDEN QUIT
930  REM
940  VTAB 19: HTAB 1: CALL  - 958
     : PRINT "GAME STOPPED WITH T
     OTAL OF";TT
950  GOTO 1260
```



```
960  REM
970  REM  WRAP UP
980  REM
990  VTAB 22: HTAB 1: PRINT "SELE
     CT TARGET ";: INVERSE : PRINT
     "TOTAL ";: NORMAL : PRINT S$
     ;: GET J$:J =  VAL (J$)
1000  VTAB 22: HTAB 1: CALL  - 86
     8: PRINT "SELECT ";: INVERSE
     : PRINT "LINE";: NORMAL : PRINT
     " NUMBER ";: GET JJ$:J =  VAL
     (JJ$)
1010 VE =  PEEK (934 + (128 * J))

1020 GT = TT * B * 10
1030  REM
1040  REM  CONFIRM WINNER
1050  REM
1060  IF TT = VE - 176 THEN  VTAB
     19: CALL  - 958: HTAB 1: INVERSE
     : PRINT "WINNER WITH ";GT;"
     POINTS": NORMAL : GOTO 1260
1070  REM
1080  REM  LOSER
1090  REM
1100 GT = 0: VTAB 19: HTAB 1: CALL
      - 958: PRINT "SORRY ...TRY
     AGAIN !"
1110  GOTO 1260
1120  REM
1130  REM  MUNCHIE NOISE
1140  REM
1150 N =  PEEK ( - 16336)
1160  RETURN
1170  REM
1180  REM  CHEATER CATCH
1190  REM
1200  POKE (935 + (128 * A)),42: VTAB
     19: HTAB 1: CALL - 958: PRINT
     "YOU PICKED A ";: INVERSE : PRINT
     "ZERO";: NORMAL : PRINT "! G
     AME OVER!";Q$;Q$;Q$: GOTO 12
     60
1210  REM
1220  REM  OVERSHOOT TRAP
1230  REM
1240  IF TT > 8 THEN  VTAB 22: HTAB
     1: PRINT Q$;Q$;Q$;"YOU ";: FLASH
     : PRINT "CAN'T";: NORMAL : PRINT
     " WIN WITH A SCORE OF ";TT;"
     !": GOTO 1260
1250  REM
1260  REM  PLAY AGAIN
1270  REM
1280  IF HS < GT THEN HS = GT
1290  PRINT "HIGH SCORE THIS SERI
     ES ";HS
1300  VTAB 23: PRINT "ANOTHER GAM
     E OF MUNCHIES?(Y/N)";B$;B$;:
      GET AN$
1310 H = H + 1
1320  IF AN$ <  > "N" THEN TT = 0
     : GOTO 330
1330  PRINT : FOR X = 1 TO 1000: PRINT
     "*";: NEXT X
1340  FOR X = 1 TO 20: VTAB 20: HTAB
     18 + X: PRINT "";: HTAB 19 +
     X: PRINT "C";: FOR Z = 1 TO
     25: NEXT Z: NEXT X
1350  FOR Z = 1 TO 1000: NEXT Z: HOME

1360  END
1370  REM
1380  REM  MUNCHIE FOOD
1390  REM
1400  VTAB RR: HTAB 2: PRINT D$;
1410  PRINT
1420  FOR F = 1 TO (4 * R) STEP 4

1430  VTAB RR: HTAB F + 4: PRINT
     "*";: GOSUB 1130: NEXT F
1440  RETURN
1450  REM
1460  REM  INSTRUCTIONS
1470  REM
1480  HOME : SPEED= 200
1490  VTAB 2: HTAB 16: INVERSE : PRINT
     "MUNCHIES": NORMAL
1500  FOR Z = 1 TO 1000: NEXT Z
1510  VTAB 4: HTAB 1: PRINT "MUNC
     HIES IS A GAME THAT WILL TAK
     E A     LIFETIME TO LEARN AN
     D A FEW MINUTES TO  PLAY!": PRINT

1520  PRINT "FOLLOW SCREN INSTRUC
     TIONS. SELECT A     NUMBER F
     OR POINTS (IN THE RIGHT COLU
     MN)."
1530  PRINT "THE OBJECT IS TO CON
     TINUE TO SELECT     NUMBERS
     WHICH WILL EQUAL THE TOTAL O
     F   THE HIGHEST POSSIBLE NUM
     BER ON THE      RIGHT. WHEN
     YOU GO AS HIGH AS YOU
     DARE -- PRESS #9..."
1550  PRINT : PRINT "THIS WILL GI
     VE YOU INSTRUCTIONS TO GET
     TOTAL. IF YOU HAVE A VALID S
     CORE, YOUR  COMPUTER WILL GI
     VE YOU THE TOTAL."
1560  PRINT : PRINT "A RECORD OF
     HIGH SCORE IS KEPT TO SEE IF
     YOU CAN BETTER ANY PREVIOUS
     SCORES."
1570  VTAB 24: HTAB 1: PRINT "===
     TO CONTINUE PREE ANY LETTER
     ===";: GET AN$
1580  SPEED= 255: RETURN
```

■

DOS指令更改器

雷眞如

本人對「電腦時代」4、5期刊登 DOS 操作指令簡化器甚感興趣，但覺得靈活性不夠。下列程式可以隨意更改 DOS 指令由一個字母至程式列出的最多可更改的字母數字，也可以還原，因此非常靈活。

未開始討論程式前必須先說明所簡化的前幾個字母（如R或RE等）不能與 DOS 或 APPLESOFT 的其他指令的前幾個字母重復（如R對 RUN 和 READ、RENAME，RE對 READ 和 RENAME），否則便只有一個指令可用。例如將 DELETE改成 DE或 DEL，則 APPLE 的 DEL 便不能用。

我們知道 28 個 DOS 指令由＄A884到＄A907 排列。在程式中把上述排列的 DOS 指令叫出，並按排列次序編上 28 個號碼，列出如下的目錄表：

1＝INIT	15＝APPEND
2＝LOAD	16＝RENAME
3＝SAVE	17＝CATALOG
4＝RUN	18＝MON
5＝CHAIN	19＝NOMON
6＝DELETE	20＝PR#
7＝LOCK	21＝IN#
8＝UNLOCK	22＝MAXFILES
9＝CLOSE	23＝FP
10＝READ	24＝INT
11＝EXEC	25＝BSAVE
12＝WRITE	26＝BLOAD
13＝POSITION	27＝BRUN
14＝OPEN	28＝VERIFY

程式接着問你是否更改指令，答Y後程式會叫你打入想更改的上述指令旁邊的號碼，打入號碼後，程式會示出打算被更改的指令名稱，並列出可更改指令名稱的字母限數。在輸入所更改的簡化或加長的名稱字母後，程式會把上述目錄表排列一次，此時你可看到你所更改的指令名稱已列入表中。■

DOS 指令更改器程式

```
1   REM *************************
2   REM * AUTO CHANGE COMMAND *
3   REM *       1984          *
4   REM *                     *
5   REM *************************
10  HOME : HTAB 8: PRINT "AUTO CH
    ANGES DOS COMMANDS": HTAB 8:
     PRINT "===================
    =====": VTAB 4: PRINT "THIS
    PROGRAM WILL CHANGE ANY DOS
    COMMANDWITH ONLY ONE LETTER
    TO SEVERAL LETTERS.": PRINT
12  PRINT "BUT, R COULD NOT STAND
     FOR RUN, RENAME  AND READ,
     OR R WILL STAND ONLY FOR RUN
      AND APPLESOFT COMMAND RUN
     CANNOT BE     USED."
14  PRINT : PRINT "ALSO, RE COULD
     NOT STAND FOR RENAME AND RE
    AD, OR RE WILL STAND ONLY FO
    R RENAME  AND READ CANNOT BE
     USED."
16  PRINT : PRINT "SAME REASON FO
    R DELETE-DEL, NOMOM-NEW,   LO
    AD-LOCK-LIST, INIT-IN#-INT,
    POSITION- PR#, CHAIN-CLOSE-C
    ATALOG, BLOAD-BSAVE-   BRUN E
    TC.                     -C.Y.
    LUI-"
20  PRINT : PRINT "PRESS <RETURN>
     TO CONTINUE ";: GET A$
30  DIM A(28),A$(28): GOSUB 200: FOR
     XX = 1 TO 28: FOR X = 0 + NN
     TO 00:Y =  PEEK (X): ON Y =
     171 GOTO 30:A$(XX) = A$(XX) +
     CHR$ (Y): ON Y > 90 GOTO 40
    : NEXT X
40 NN = NN +  LEN (A$(XX)): NEXT
    XX:AB = 132 - NN
50  HOME : INVERSE : VTAB 1: HTAB
    14: PRINT "DOS COMMANDS": NORMAL
    : GOSUB 60: GOTO 70
60  FOR X = 1 TO 14: VTAB X + 2: HTAB
    7: PRINT X;: HTAB 10: PRINT
    "= ";A$(X);: HTAB 22: PRINT
    X + 14;: HTAB 25: PRINT "= "
    ;A$(X + 14): NEXT : RETURN
70  VTAB 18: PRINT "DO YOU WANT T
    O CHANGE DOS COMMAND?(Y/N)";
    : GET Z$: ON Z$ = "N" GOTO 1
    80
75  VTAB 18: HTAB 1: INPUT "WHICH
     COMMAND DO YOU WANT TO CHAN
    GE ?    (ENTER ITS NO.) ";Z$:
    Z =  VAL (Z$): IF Z < 1 OR Z
```

DOS 指令更改器

```
    > 28 THEN 75
77   VTAB 23: PRINT "MAX. LETTER N
      O. FOR NEW COMMAND : ";: FLASH
      : PRINT AB +  LEN (A$(Z)): NORMAL
80   VTAB 19: HTAB 19: PRINT "--IN
      STEAD OF ";: FLASH : PRINT A
      $(Z): NORMAL : INPUT "ENTER
      NEW COMMAND : ";N$: ON N$ =
      "" GOTO 150:A$(Z) = N$
90    ON Z = 1 GOTO 100: FOR X = 1 TO
      Z - 1:A(X) =  LEN (A$(X)): FOR
      Y = 1 TO A(X):N = N + 1: NEXT
      : NEXT
100   HOME : FOR X = Z TO 28:A(X) =
       LEN (A$(X)): FOR Y = 1 TO A
      (X):N = N + 1:A =  ASC ( MID$
      (A$(X),Y,1))
110   IF Y = A(X) AND A < 91 THEN
      A = A + 128: POKE N,A: PRINT
       CHR$ (A);: HTAB 30: PRINT N
      : GOTO 130
120   POKE N,A: PRINT  CHR$ (A);: IF
      Y = A(X) THEN  HTAB 30: PRINT
      N
130  NEXT : NEXT : IF OO - N > O THEN
      FOR X = N + 1 TO OO: POKE X
      ,171: PRINT "AB";" ";: NEXT
      :AB = OO - N
150  GET Z$: HOME : FLASH : VTAB
      1: HTAB 12: PRINT "NEW";: INVERSE
      : PRINT " DOS COMMANDS": NORMAL
      : GOSUB 60
160  VTAB 18: PRINT "DO YOU WANT
      TO CHANGE OTHER DOS COMMAND
      ? (Y/N) ";: GET Z$: IF Z$ <
       > "Y" AND Z$ <  > "N" THEN
      POKE 34,16: HOME : POKE 34,
      0: GOTO 160
170  IF Z$ = "Y" THEN  POKE 34,16
      : HOME : POKE 34,0: GOSUB 20
      0: GOTO 75
180  END
200 N =  PEEK (978) * 256 + 2947:
      O = N + 1:OO = N + 132: RETURN
      : REM O=43140($A884),
           OO=43271($A907)
```


1·2·3 COMPUTER CENTRE 電腦中心

〈專營〉

多款 電腦硬件，軟件，配件，零件，盒帶，磁碟，書籍，保養及維修。

・收購・

二手電腦，任何牌子，磁碟機，顯示器或其他 附件一律歡迎，全港最高，不妨試試。

（歡迎查詢）



—特設—

訂造 JOYSTICK，款式大小任君選擇，適合任何電腦，比普通耐用十倍，緊張刺激，仿似身在遊戲中心一樣。保用180天。

代客 訂購名廠原裝 IBM-APPLE II-MICROSOFT 電腦及軟件

apple

地址：深水埗南昌街白田購物中心一樓 101，102 室（南昌戲院後面）
電話：3-7783923

光速計數器得獎改良程式

上期刊出之光速計數器的計數進位速度被許多讀者打破了，他們設計出來的修訂程式速度均較原程式爲快，其中更有幾個非常優異的時間——由0數至一百萬只需約七秒；由0數至一億不足十一分。

事實上，原程式在設計上已相當精簡，在程式長度方面，却未能有加以突破，一般的參加者程式均較原程式長。當然，爲追求速度，這方面是可以容忍的。有一位讀者曾透露他有一個程式比我們的得獎者的七秒多速度還要快，不過程式長度則約爲 15 個 SECTOR！

得獎程式是由朱永昌君（上水馬會道）所提交的。程式不單在速度上加快了多倍，而且可將8位數字顯示更改爲9位，使可以顯示一億的數目。

朱永昌君可獲贈十二期電腦時代雜誌及精選程式磁碟一隻。未獲入選的參加者要再接再勵，參加今期的各項遊戲。

得獎程式（朱永昌）

貴刊第五期曾刊載一「光速計數器」的程式，程式很能巧妙利用到分支及迴圈，而且全式精簡，因此鄙人對筆者也甚爲佩服；唯筆者認爲很難找著一個程式所數數目的速度比它還快，但我却不以爲然，須知記憶體和時間成反比，如能適當減少迴圈，程式雖會因此加長，但却必能減短時間；本人對編寫程式也很有趣，故特此寫成令一個「光速計數器」的程式，願爲指敎。

程式特點：

1、由0數至 1,000,000大約7秒，爲原程式的三倍。

2、由0數至 100,000,000 不足 11 分。

3、全程式長度約爲原式 4.5 倍。■

```
*300L

0300-    20 58 FC    JSR    $FC58
0303-    A9 B0       LDA    #$B0
0305-    A2 09       LDX    #$09
0307-    9D B7 05    STA    $05B7,X
030A-    CA          DEX
030B-    D0 FA       BNE    $0307
030D-    A0 B1       LDY    #$B1
030F-    A2 B0       LDX    #$B0
0311-    4C 17 03    JMP    $0317
0314-    EE BF 05    INC    $05BF
0317-    8C C0 05    STY    $05C0
031A-    EE C0 05    INC    $05C0
031D-    EE C0 05    INC    $05C0
0320-    EE C0 05    INC    $05C0
0323-    EE C0 05    INC    $05C0
0326-    EE C0 05    INC    $05C0
0329-    EE C0 05    INC    $05C0
032C-    EE C0 05    INC    $05C0
032F-    EE C0 05    INC    $05C0
0332-    8E C0 05    STX    $05C0
*L

0335-    AD BF 05    LDA    $05BF
0338-    C9 B9       CMP    #$B9
033A-    D0 D8       BNE    $0314
033C-    8E BF 05    STX    $05BF
033F-    AD BE 05    LDA    $05BE
0342-    C9 B9       CMP    #$B9
0344-    F0 06       BEQ    $034C
0346-    EE BE 05    INC    $05BE
0349-    4C 17 03    JMP    $0317
034C-    8E BE 05    STX    $05BE
034F-    AD BD 05    LDA    $05BD
0352-    C9 B9       CMP    #$B9
0354-    F0 06       BEQ    $035C
0356-    EE BD 05    INC    $05BD
0359-    4C 17 03    JMP    $0317
035C-    8E BD 05    STX    $05BD
035F-    AD BC 05    LDA    $05BC
0362-    C9 B9       CMP    #$B9
0364-    F0 06       BEQ    $036C
0366-    EE BC 05    INC    $05BC
*L

0369-    4C 17 03    JMP    $0317
036C-    8E BC 05    STX    $05BC
036F-    AD BB 05    LDA    $05BB
0372-    C9 B9       CMP    #$B9
0374-    F0 06       BEQ    $037C
0376-    EE BB 05    INC    $05BB
0379-    4C 17 03    JMP    $0317
037C-    8E BB 05    STX    $05BB
037F-    AD BA 05    LDA    $05BA
0382-    C9 B9       CMP    #$B9
0384-    F0 06       BEQ    $038C
0386-    EE BA 05    INC    $05BA
0389-    4C 17 03    JMP    $0317
038C-    8E BA 05    STX    $05BA
038F-    AD B9 05    LDA    $05B9
0392-    C9 B9       CMP    #$B9
0394-    F0 06       BEQ    $039C
0396-    EE B9 05    INC    $05B9
0399-    4C 17 03    JMP    $0317
039C-    8E B9 05    STX    $05B9
*L

039F-    8C B8 05    STY    $05B8
03A2-    60          RTS
03A3-    1C          ???
03A4-    1D 1E FF    ORA    $FF1E,X
```

優異程式（謝偉機）

閱貴刊第五期“光速計數器”一文，甚感興趣，寄上一段程式請教，此程式速度較之原文的快 2·3倍，原因是在個位增值時不採用檢測，而直接加至“9”，節省不少時間，但此程式與原文之程式均無實用，因爲在進位與不進位時之數數時間不同，只是作爲玩意。

由於工作需要，本人在檢修高速磁帶時，經常採用類似計數方法，以中斷觸發計數程式，配合 RAM之位址使中斷進入點與計數值同步，以 PLA 取回 RAM之位址，則等於每次測量之數值，以 CPU 6502而言，每加1需時2 US，計數之最大值與 RAM之容量有關，誤差亦是1（2 US），如有興趣可互相研究一下，不過無論如何，總不如採用硬件綫路的好。□

```
*300L

0300-   20 58 FC   JSR   $FC58
0303-   A9 B0      LDA   #$B0
0305-   A2 08      LDX   #$08
0307-   9D B8 05   STA   $05B8,X
030A-   CA         DEX
030B-   D0 FA      BNE   $0307
030D-   A2 08      LDX   #$08
030F-   FE B8 05   INC   $05B8,X
0312-   FE B8 05   INC   $05B8,X
0315-   FE B8 05   INC   $05B8,X
0318-   FE B8 05   INC   $05B8,X
031B-   FE B8 05   INC   $05B8,X
031E-   FE B8 05   INC   $05B8,X
0321-   FE B8 05   INC   $05B8,X
0324-   FE B8 05   INC   $05B8,X
0327-   FE B8 05   INC   $05B8,X
032A-   A9 B0      LDA   #$B0
032C-   9D B8 05   STA   $05B8,X
032F-   CA         DEX
0330-   F0 0D      BEQ   $033F
*L

0332-   A9 B9      LDA   #$B9
0334-   DD B8 05   CMP   $05B8,X
0337-   F0 F1      BEQ   $032A
0339-   FE B8 05   INC   $05B8,X
033C-   4C 0D 03   JMP   $030D
033F-   60         RTS
0340-   00         BRK
```

優異程式（黃贊輝）

隨函附上一個較快的計算器程式，由1數到一百萬約需7.3秒，而且1數到 99,999,999 ，程式只用了十二分十秒，其實，整個程式的結構跟第五期刊出來的沒有分別，只不過是省去了一些迴路。

第一部份由 300—312 是用作清除畫面。

第二部份由 313—32C 是用作設定8個零。

第三部份由 32D—34C 是用作計算個位由0至9。

第四部份由 34D—3B1 時用作其它位置的處理。

整個程式最能節省時間的是第三部份只用去了 30 個步驟去完成個位0至9，而第五期的程式却用上了超過1倍以上的時間。

雖然所需時間是少了，但却令到程式增長了好幾倍。□

```
0300-   A9 A0      LDA   #$A0
0302-   A2 00      LDX   #$00
0304-   9D 00 04   STA   $0400,X
0307-   9D 00 05   STA   $0500,X
030A-   9D 00 06   STA   $0600,X
030D-   9D 00 07   STA   $0700,X
0310-   CA         DEX
0311-   D0 F1      BNE   $0304
0313-   A2 B0      LDX   #$B0
0315-   8E C0 05   STX   $05C0
0318-   8E BF 05   STX   $05BF
031B-   8E BE 05   STX   $05BE
031E-   8E BD 05   STX   $05BD
0321-   8E BC 05   STX   $05BC
0324-   8E BB 05   STX   $05BB
0327-   8E BA 05   STX   $05BA
032A-   8E B9 05   STX   $05B9
032D-   A9 B9      LDA   #$B9
032F-   EE C0 05   INC   $05C0
0332-   EE C0 05   INC   $05C0
0335-   EE C0 05   INC   $05C0
0338-   EE C0 05   INC   $05C0
033B-   EE C0 05   INC   $05C0
033E-   EE C0 05   INC   $05C0
0341-   EE C0 05   INC   $05C0
0344-   EE C0 05   INC   $05C0
0347-   EE C0 05   INC   $05C0
034A-   8E C0 05   STX   $05C0
034D-   CD BF 05   CMP   $05BF
0350-   F0 06      BEQ   $0358
0352-   EE BF 05   INC   $05BF
0355-   4C 2F 03   JMP   $032F
0358-   8E BF 05   STX   $05BF
035B-   CD BE 05   CMP   $05BE
035E-   F0 06      BEQ   $0366
0360-   EE BE 05   INC   $05BE
0363-   4C 2F 03   JMP   $032F
0366-   8E BE 05   STX   $05BE
0369-   CD BD 05   CMP   $05BD
036C-   F0 06      BEQ   $0374
036E-   EE BD 05   INC   $05BD
0371-   4C 2F 03   JMP   $032F
0374-   8E BD 05   STX   $05BD
0377-   CD BC 05   CMP   $05BC
037A-   F0 06      BEQ   $0382
037C-   EE BC 05   INC   $05BC
037F-   4C 2F 03   JMP   $032F
0382-   8E BC 05   STX   $05BC
0385-   CD BB 05   CMP   $05BB
0388-   F0 06      BEQ   $0390
038A-   EE BB 05   INC   $05BB
038D-   4C 2F 03   JMP   $032F
0390-   8E BB 05   STX   $05BB
0393-   CD BA 05   CMP   $05BA
0396-   F0 06      BEQ   $039E
0398-   EE BA 05   INC   $05BA
039B-   4C 2F 03   JMP   $032F
039E-   8E BA 05   STX   $05BA
03A1-   CD B9 05   CMP   $05B9
03A4-   F0 06      BEQ   $03AC
03A6-   EE B9 05   INC   $05B9
03A9-   4C 2F 03   JMP   $032F
03AC-   8E B9 05   STX   $05B9
03AF-   4C 2F 03   JMP   $032F
03B2-   00         BRK
```

第五期問題棋盤答案

第五期所刊出的問題棋盤錯誤之處是棋盤的格子顏色調弄錯了。國際象棋棋盤左上角一格應爲黑色，而非白色，最簡單的答案如下：將行號

```
30 COLOR＝0：IF（X＋L）／10＜＞
   INT（（X＋L）／10）THEN COLOR＝15
```

改爲：

```
30 COLOR＝15：IF（X＋L）／10＜＞INT
   （（X＋L）／10）THEN COLOR＝0
```

我們非常意外，收到出乎意料之數目的參加者，大部份都能答中，不過亦有部份參加者可能不熟習國際象棋的棋盤，而將注意力放在程式結構中，忽略了問題的主要關鍵。然而，這亦使我們感到高興，許多讀者對 BASIC 語言有一定程度的水準，可以在程式結構方面找到改進之處。

鑑於篇幅所限，我們未能將猜中者的改良程式一一刊登，只能抽選其中幾個較爲有優異的答案刊登。未有獲獎的參加者不要灰心，請再接再勵來信參加。

今期得獎的三位參加者：

（1）李用智（北角英皇道）
（2）張建生（柴灣）
（3）趙仕培（坪石邨）

各人可獲贈閱十二期電腦時代雜誌（由第七期開始）

□

得獎答案(1)李用智

本人覺得有以下的地方不妥善。

1、指令的設定及變數的設定如分爲兩行便更容易使人明白及參考。

2、第 10 行之 HOME 指令根本沒有作用，因 GR 指令一定會把畫面自動清除及設定最低的 4 行爲文字顯示。

3、因本程式是繪畫國際象棋棋盤，而 GR 這指令的畫圖大小是縱 40 行，橫 40 列。因此畫出的棋盤下半部之底 2 行將會是電腦操作者的放棋位置。而國際象棋規例中，下棋者的右下角應是白方格。於是此程式的黑白方格，便反轉了。

4、在第 60 行中，Y＝0 此敍述沒有用處。因緊接其後的指令是 NEXT X，而X之 FOR LOOP（第 20 行）又將Y重新設定爲L之値。

5、指令 END 應是一程式最尾之敍述，所以行 70 應予更改。

所以更改後之程式如下：

```
10  TEXT : GR
20 L = 0
30  FOR X = 0 TO 35 STEP 5:Y = L
40  COLOR= 15: IF (X + Y) / 10 <
     >  INT ((X + Y) / 10) THEN
     COLOR= 0
50  FOR Y = L TO L + 4
60  HLIN X,X + 4 AT Y
70  NEXT Y: NEXT X
80 L = L + 5: IF L <  = 35 THEN 3
    0
90  END
```

□

得獎答案(3)趙仕培

程式中，行號 60 中的Y＝0 是非必要的，因爲在它之後的 NEXT X 迴圈到 20 便立刻設定Y＝L，雖然要仍無妨，但這是會增加程式的複雜性。

另外，行號 20 的Y變數（計算顏色的參數值）和行號 40 的Y變數（決定垂直坐標位置的數到）都利用同一變數表示，在這裡，雖然仍對程式的運行無阻，但很容易便把它們混淆，所以小弟認爲把20 行中的Y變數盡可能改爲其他變數，例如A，B；但小弟認爲最好便是把行 20 中的Y變數除去，在 30 行中直接代入L變數，使程式更加容易閱讀了。

```
20 FOR X＝0 TO 35 STEP 5
30 COLOR＝O：IF（X＋L）／10＜＞
   INT（（X＋L）／10）THEN COLOR＝15
```

最後一點，也是小弟最覺得奇怪的一點，小弟對國際象棋認識甚淺，但知道一般的標準棋盤的右下角的格是塗白（COLOR＝15），但這程式棋盤是塗黑的（COLOR＝0），若是左右對奕便無問題，但通常都是上下對奕的，若寫成完整的程式，便應改爲上下對奕，這使玩者更有投入感。

綜合以上各點，修改後的程式如下：

```
10  TEXT : HOME :L = 0: GR
20  FOR X = 0 TO 35 STEP 5
30  COLOR= 0: IF (X + L) / 10 =  INT
     ((X + L) / 10) THEN  COLOR=
     15
40  FOR Y = L TO L + 4
50  HLIN X,X + 4 AT Y
60  NEXT Y,X
70 L = L + 5: IF L > 35 THEN  END
80  GOTO 20
```

□

得獎答案(2)張建生

我在閱讀貴刊第五期的捉蟲訓練，對程式有以下的意見，其實，在行 20 的Y＝L和 60 行的Y＝0 是多餘的，程式中的行 30 的Y轉爲L，程式如下：

```
10  TEXT : HOME :L = 0: GR
20  FOR X = 0 TO 35 STEP 5
30  COLOR= 0: IF (X + L) / 10 < >
     INT ((X + L) / 10) THEN  COLOR=
    15
40  FOR Y = L TO L + 4
50  HLIN X,X + 4 AT Y
60  NEXT Y: NEXT X
70 L = L + 5: IF L > 35 THEN  END

80  GOTO 20
```

（程式一）

```
10  TEXT : HOME : GR
20  FOR Y = 0 TO 35 STEP 5
30  FOR X = 0 TO 35 STEP 5
40  COLOR= 15: IF (X + Y) / 10 <
    >  INT ((X + Y) / 10) THEN
    COLOR= 0
50  FOR L = Y TO Y + 4
60  HLIN X,X + 4 AT L
70  NEXT : NEXT : NEXT :
```

（程式二）

程式一是根據原文所改的，但這個程式太沒有結構了，筆者作了程式二，這個程式的效果和原文一樣。但是，黑白顏色已經轉回正確的棋盤。

通常棋盤的方格是比較正方的，不似得原程式的長方型，所以筆者作了程式三。

```
10  TEXT : HOME : GR
20 C = 15: COLOR= C
30  FOR Y = 0 TO 35 STEP 5
40  FOR X = 0 TO 28 STEP 4
50  FOR L = Y TO Y + 4
60  HLIN X,X + 3 AT L
70  NEXT L
80 C =  ABS (C - 15): COLOR= C
90  NEXT X
100 C =  ABS (C - 15): COLOR= C
110  NEXT Y
```

程式四是在文字幕做的，它不但繪出棋盤。而且還有每一格的座標。

這 PROGRAME 只是給編輯先生參考的。

（程式四）

```
10  TEXT : HOME :A$ = "  "
20  FOR Y = 4 TO 18 STEP 2
30  FOR X = 4 TO 32 STEP 4
40  NORMAL :C =  ABS (C - 1): IF
     C THEN  INVERSE
50  VTAB Y: HTAB X: PRINT A$: VTAB
     Y + 1: HTAB X: PRINT A$
60  NEXT X
70 C =  ABS (C - 1)
80  NEXT Y
85  NORMAL
90  FOR Y = 4 TO 18 STEP 2
100  HTAB 36: VTAB Y: PRINT  CHR$
     (47 + Y / 2)
110  NEXT Y
120  FOR X = 5 TO 33 STEP 4
130  HTAB X: VTAB 20: PRINT  CHR$
     (64 + X / 4)
140  NEXT X
150  END
```

□

優異答案(1)黃兆才

這個程式雖然是繪劃一個國際象棋的棋盤，但是它所顯示的格子顏色剛巧與正確的國際象棋棋盤相反。換句說話，正確棋盤的格子是白色的，它却顯示爲黑色；而正確棋盤的格子是黑色的，它却顯示爲白色。這便是棋盤不妥之處，只要將程式中行號 30 內的〝< >〞改爲〝=〞，問題即可解決。

另一方面，在行號 20 內的Y＝L及行號 60 內的Y＝0 都是可以不要的，只需將行號 30內的Y改爲L便可。

經修改後爲程式便能顯示正確的國際象棋棋盤，程式如下：

```
10  TEXT : HOME :L = 0: GR
20  FOR X = 0 TO 35 STEP 5
30  COLOR= 0: IF (X + L) / 10 =  INT
     ((X + L) / 10) THEN  COLOR=
     15
40  FOR Y = L TO L + 4
50  HLIN X,X + 4 AT Y
60  NEXT Y: NEXT X
70 L = L + 5: IF L > 35 THEN  END

80  GOTO 20
```

或者將程式再修改如下，亦可得到同樣效果。

```
10  TEXT : HOME : GR
20  FOR L = 0 TO 35 STEP 5
30  FOR X = 0 TO 35 STEP 5
40  COLOR= 0: IF (X + L) / 10 =  INT
     ((X + L) / 10) THEN  COLOR=
     15
50  FOR Y = L TO L + 4
60  HLIN X,X + 4 AT Y
70  NEXT Y,X,L
```

□

第1至4期程式磁碟總目錄

1

	附有程式之文章	相對檔案名稱
1	突破傳統的鍵盤	DVORAK KEYBOARD
2	捉球遊戲	BEND OF ROUNDING
3	勇闖電腦室	GAME OF SABOTAGE
4	迷宮救美	GAME OF MAZING
5	將 APPLESOFT	MAZE SHAPES
	變爲 FORTRAN	APPLESOFT TO FORTRAN
6	上、下、左、右移	PEELING THE TEXT SCREEN
	動文字畫面	PEELING LISTING 5
7	測試工具— BOO—	BOOLEAN TUTORIAL
	LEAN 程式	BOOLEAN TUTORIAL—MAIN PROGRAM
8	騰空更多磁碟空間	GREETING TRANSFER
	一搬移HELLO	GREETING
9	重現刪除之檔案	RETRIEVE THE DELETED PROGRAM
10	電腦時代月曆	APPLE CALENDAR
		APPLE CALENDAR & SUM
		SUM 1
		CAL.FILE—1983
		CAL.FILE—1984
		CAL.FILE—1985
		CAL.FILE—1986
11	COMPUTEXT 文字處理	COMPUTEXT
12	時日相距計算程式	DYNAMIC CALENDAR
13	增強英文字滙能力	GAME OF SPELLING TEST
14	幼童數學四則運算練習	GAME OF MATHEMATIC
15	過四關	GAME OF CONNECT FOUR

2

	附有程式之文章	相對檔案名稱
1	貨櫃處理	CONTAINER HANDLING
2	沙蟹幻方局	POKER
		LOADER
		POKER SHOWHAND
3	巧破密令	GAME OF CRYPTO
4	美術字繪劃器	RETRIEVE FONTASTIC
		SCROLL!!
		SUPERFONTASTIC
		LEAD
		CHAR 1
5	小階字母處理	LOWER CASE CHIP CONVERTER
6	程式轉載器	SAFELY HANDLING PROGRAM
7	彩色電視/監視器色彩測試	CRT COLOR DEMONSTRATION TEST
8	保留遊戲積分記錄	HIGH SCORE KEEPER
9	數學套件程式	MATHS TOOL KIT 1
		Z
10	磁碟修護工具	BPHTCH
		PATCH 2
11	加設電腦時代月曆	LOADSUM 2
	圖像一個	SUM 2

3

	附有程式之文章	相對檔案名稱
1	DOS 的自動同步追踪器	A DOS SELF-SYNCH DEMO
2	動量不滅定律模擬法	AIR CAR - MOMENTUM
3	自製磁碟索引表封套	APPLE DISKETTE ENVELOPE
4	蘋果編曲器	APPLE MUSIC WRITER
		PROGRAM#1-SOUND & VARIATION
		PROGRAM#2-PADDLE SOUND
		PROGRAM#3.OK
		PROGRAM#4.OK
		PROGRAM#5.OK
5	幼童認數字初階	BOUNCING NUMBER-MAIN PROGRAM
		MSOUND
		NUMBER TABLE
6	賽跑遊戲	PATHWAY
7	步步追踪	CHASE
8	DOS KILLER	DOS KILLER
9	圍棋遊戲	DOT
10	巴哈的古典樂章名作	FUNGUE
11	旋轉螢幕畫面	LEAVING HOME
		LEAVING HOME DEMO
12	雜誌目錄分類歸檔系統	MAGAZINE FILE TRACKING SYSTEM
13	神經病醫生一神經鬆弛體操	RELAX ROUTS
		RELAX TEXT
		RELAXATION-MAIN PROGRAM
		CREATE RELAX FILES

4

	附有程式之文章	相對檔案名稱
(1)	DOS 操作指令簡化器	COM CHANGE
(2)	TEXT FILE 有些甚麼內容?	TEXT WATCHER
(3)	立體迷宮	ICE CAVES
(4)	英文串字練習	BEAR 1
		BEAR 2
		BOUNCING WORD
		TOOTLE
(5)	如何以機械語言來修補 DOS	DOS MONZAPPER
(6)	電腦動畫製作原理	ASMINPUT
		TEST F
		TEST G
		TEST H
		MANC
		SEQUENCE CREATOR2
		LEAD SEQUENCE
(7)	豐富設計的學前兒童教育程式	KINDERGARTEN
(8)	應用數學套件程式(二)	MATH TOOLKIT2
(9)	DOS WRITER	DOS WRITER
		EXAMPLE
		EXAMPLE 2
		ORIGINAL EXAMPLE
		ORIGINAL EXAMPLE 2
		TEXT
(10)	穿梭機哥倫比亞號機	SPACEDUST
(11)	找回失落了的 STEP 及 TRACE	STEP-TRACE
		STEP-TRACE 2
(12)	TRACER控制常式	TRACER
(13)	將 RAM 卡作爲磁碟機使用	LEAVE RAMDISK
		MAKE EMUDISK
		RAM DISK
		RAM DISK TEXT
(14)	黑房計時器	TIMER
(15)	太陽系模擬系統	SOLAR SYSTEM

請前往COMPUHOUSE補購

佐敦廣東道582號廣發商業大厦八樓三室。TEL :3-312007

領先 Apex 助你掌握電腦！



私人及小組電腦証書課程

1. Programmer's Full Cert. Training Course
2. Computer Operation & Basic Programming
3. Advanced Basic
4. Disk Operating System (File Handling)
5. Wordstar and CP/M Operating System
6. dBase II and Utility Packages
7. Assembly Language, Cobol, Logo, Fortran or Pascal
8. Peachtree Inventory Control System
9. Peachtree Account System
10. 電腦磁碟折銷，保護及原理
11. 初級電腦裝嵌及修理
12. 初級兒童電腦操作及認識
13. 中文電腦操作，會考電腦補習

私人教授，天天開課。小組學習，興趣濃厚

領先私人電腦訓練中心　Apex Computer Training Centre

＊歡迎報名或電話查詢＊

地址：九龍深水埗青山道135號3樓11室（近九江街，金盟商場對面，電梯2字）
電話：3-7280889，3-7299187

電腦公司及製造商請注意！

貴　公司是否希望將最新的電腦產品訊息傳遞給電腦用家知曉？那麼爲何不選擇一本擁有高消費能力兼廣泛層面讀者基礎的雜誌。

電腦時代就是貴　公司最適當的選擇。我們擁有廣泛層面的讀者。他們不但擁有高消費能力，亦是標準的微電腦用家。他們渴望知道貴　公司所出品或代理的最新產品訊息。敬請立即與本刊廣告部接洽。

電腦時代廣告部電話3-312007

香港電腦軟件公司

ROM & RAM computer ltd

582–592, Nathan Road, Basement Shop No. B35 Sino Centre, Kowloon.
Tel: 3-306839

首創

24 小時

電話熱線服務

專業提供微型電腦之上門保養及維修服務

凡曾在下列商戶選購之電腦及其配件均可獲本公司之特價優待（保養、維修）。

東成無線電行
聯豐電腦中心
黃金電腦公司
海洋電腦公司
精確電腦公司
亞洲電腦公司
榮來影音公司
利達電腦公司
泛美電子有限公司
志成洋行電腦部
良然有限公司
菲利圖有限公司
金華電腦
商用電腦系統公司
信和電腦公司
和怡電腦公司

HONG KONG COMPUTER CO.
B. M. ELECTRONIC CO., LTD.
YEN ENTERPRISE CO., LTD.
CAPITAL COMPUTER CO.
HIP SHING COMPUTER CO.
JIM MY ELECTRONIC CO.
BILON PHOTO & HI-FI CENTRE
COMPUTER TECHNOLOGY CO.
MICUSARE CO., COMPUTER SUPPLY
MODERN ELECTRONIC CO.
IS INTERNATIONAL SCIENTIFIC CO.
MODERN ELECTRONICS COMPUTER CO.
CHIN HSIN INDUSTRIAL CO.
B & R ELECTRONIC SYSTEM (H. K.) LTD.
COMPUTER SERVICES CENTRE CO.
ELECTRONIC TECHNOLOGY CO.
BETA ELECTRONIC CO.
SUN SING TOYS CO.
COLD COMPUTER CO.
NAM HUNG ELECTRONIC CO.

（排名不分先後）

歡迎垂詢
熱線電話
(3-306839)

COMPUTING AGE

電腦時代

第六期
一九八四年五月十五日出版
本刊每月十五日出版
每本訂價港幣八元正

編者註：

（1）今期刊出之「APPLESOFT 擴展版本」在各項指令上增添強大功能。讀者會發覺用新指令編寫程式會改快，更容易。

（2）「APPLESOFT 語法偵錯系統」可以幫助讀者在打入 APPLESOFT 程式後作打字偵錯，雖然此系統並不能檢查程式中的變數，但一般的指令，語法等皆能在最短時間內自動檢測出。

（3）「妙用無窮的 128 RAMCARD」詳細講述 128 K RAMCARD 的功能，讀者實不容錯過。

電腦時代第六期的出版日期又再一次延誤，令愛護我們的讀者久候，我們實在感到萬分歉意。電腦時代自出版以來每一期都未能準時出版，除各項客觀環境（例如假期等）影响外，最主要的還是來自編輯部。鑑於編輯部在製訂內容方面，採取嚴謹態度，力求在交往印刷廠付印之前，必須再三審閱，務使將錯誤發生率減至最低。特別在軟件程式方面。而且，我們盡量要使文章內容深入淺出，行文流暢，文字可讀性高，此舉令編輯部壓力非常之大。不過，經過上月大事改革，增聘人手後，我們有足夠的信心，今後電腦時代不會再脫期。

請記住，電腦時代的出版日期將是每月的十五日。

出　　版/ 電腦時代出版社
地　　址/ 廣東道582號廣發商業大厦703室
編　　輯/ 電腦時代編輯部
通訊地址/ 九龍中央郵箱71193號
通訊電話/ 3-312007
軟件支援/ CompuHouse
讀者服務部/ 廣東道582號廣發商業大厦八樓三室
承　　印/ 永恒印刷公司
葵涌工業街美安工業大廈12字樓A座
發　　行/ 同德書局
九龍西洋菜街193號

今期內容

□ Program saved on disk

特稿

機械人特輯
- 機械人控制語言 41
- 機械人的定義及種類 46
- 機械人的基本認識 48

妙用無窮的 128 K RAM 咭
- DOS 擬磁碟 58
- CP/M 擬磁碟 60
- PASCAL 擬磁碟 62
- 64 K 及 128 K RAM 咭的運用 63

應用軟件庫
- □ APPLESOFT 擴展版本 8
- □ 私人會計系統 20
- □ APPLESOFT 程式語法偵錯工具 33

專欄
- APPLE PASCAL 專欄 啟動方法及程式結構 54
- CP/M 專欄 檔案管理 52
- DOS3.3 專欄 DOS 緩衝器 56

蘋果工具箱
- □ 數基轉換器 34
- □ GO][一個取代 HELLO 的程式 36
- □ 完整 CATALOG 50
- □ DOS 指令再修訂版本 82

兒童電腦教育
- □ 神奇數學方塊 66
- □ 砌積木 68

電子歡樂圈
- □ 迷宮搜秘 71
- □ 太空侵略戰 74
- □ 亡命飛車 78
- □ 累加遊戲 80

精品店
- □ 爲磁碟命名 39
- □ 任意定義鍵盤 50
- □ 捉蟲訓練問題 53
- 讀者天地（上期問題棋盤答案及超光速計數器）86

其他
- □ 布倫時鐘 40
- 特惠優待讀者證申請辦法 3　訂閱 90
- 蘋果宣佈推出最新便携式２Ｃ型電腦 55
- 一九八四年青年科學家選舉 35
- 服務部通訊欄 89
- 第1至第5期程式磁碟總目錄 88　更正 35

電腦時代程式磁碟

每張只售HK$29.90

電腦時代每期都刊登多個精彩的程式，其中必然有你喜歡的。當你讀完文章興緻勃勃的坐下來欲進一步深入了解程式的精彩處時，看見那大堆冗長的列表，就甚麼興趣也沒有了！那麼，你爲何不購買一張「電腦時代程式磁碟」呢！

每期出版的電腦時代所登載的程式，均全部紀錄入一張高品質的磁碟上，並隨雜誌同時出版。爲省卻打入程式的麻煩，徒然浪費你寶貴的時間，免除了花精神捉錯打字的痛苦，今日就請就近的特約經銷處或電腦時代讀者服務部/COMPUHOUSE購備一張。

程式磁碟每張只售港幣 29.90 元。

第6期程式磁碟内容：

*All Programs are in Apple version

(1) 私人會計系統。
(2) APPLESOFT 擴展版本。
(3) GO][，一個取代 HELLO 的程式。
(4) APPLESOFT 偵錯系統。
(5) DOS 指令再修訂版本。
(6) 布倫時鐘。
(7) AMPERSAND 數基轉換器。
(8) 完整索引錄。
(9) 砌積木。
(10) 神奇數字方塊。
(11) 亡命飛車。
(12) 迷宮搜秘。
(13) 太空怪侵略戰。
(14) 累加遊戲。
(15) 爲磁碟命名。
(16) 任意定義鍵盤。
(17) 捉蟲訓練問題。

以下爲各特約經銷處地址：

百老滙冲印公司

Shop 1	**758A, Nathan Rd., Shop 8, G/F., Kowloon.**	**Tel: 3-816931**
Shop 2	**79, Broadway St., Stage IV, Mei Foo Sun Chuen, Kln.**	**Tel:3-7424923**
Shop 3	**746-8 Nathan Road, Kowloon.**	**Tel: 3-943827**
Shop 4	**3 Mut Wah St., Kwun Tong, Kowloon.**	**Tel: 3-419513**
Shop 5	**3, Queen Victoria St., Central, Hong Kong.**	**Tel: 5-235513**
Shop 7	**Shop 232 Cityplaza Taikoo Shing, Hong Kong.**	**Tel: 5-672006**
Shop 8	**207, Cheung Sha Wan Rd., Shamshuipo, Kowloon.**	**Tel: 3-796079**
Shop 9	**26, G/F., Soy Street, Mongkok, Kowloon.**	**Tel: 3-884331**
Shop 10	**60B, G/F., Argyle St., Mongkok, Kowloon.**	**Tel: 3-956367**
Shop 11	**230A, Sha Tsui Rd., Tsuen Wan, N.T.**	**Tel: 0-438926**
Shop 12	**35 G/F., Tung Yan St., Kwun Tong, Kowloon.**	**Tel: 3-449940**
Shop 13	**148 Ground Floor, Castle Peak Road, Kowloon.**	**Tel: 3-795938**
Shop 14	**529, Hennessy Road, Causeway Bay, H.K.**	**Tel: 5-770181**

亞當影音

ADAM CAMERA	**28A, Stanley Street, Central, Hong Kong.**	**Tel: 5-240723**
ADAM CAMERA	**Pioneer Building, G/F., 748A, Nathan Road, Kowloon.**	**Tel: 3-940526**

電腦時代讀者服務部/ CompuHouse

地址：九龍廣東道583-588號廣發商業大厦八樓三室
（近佐敦道渡船閣） 電話：3-312007

補購各期程式磁碟可前往讀者服務部／COMPU HOUSE

Apple is the trademark of Apple Computer Inc.